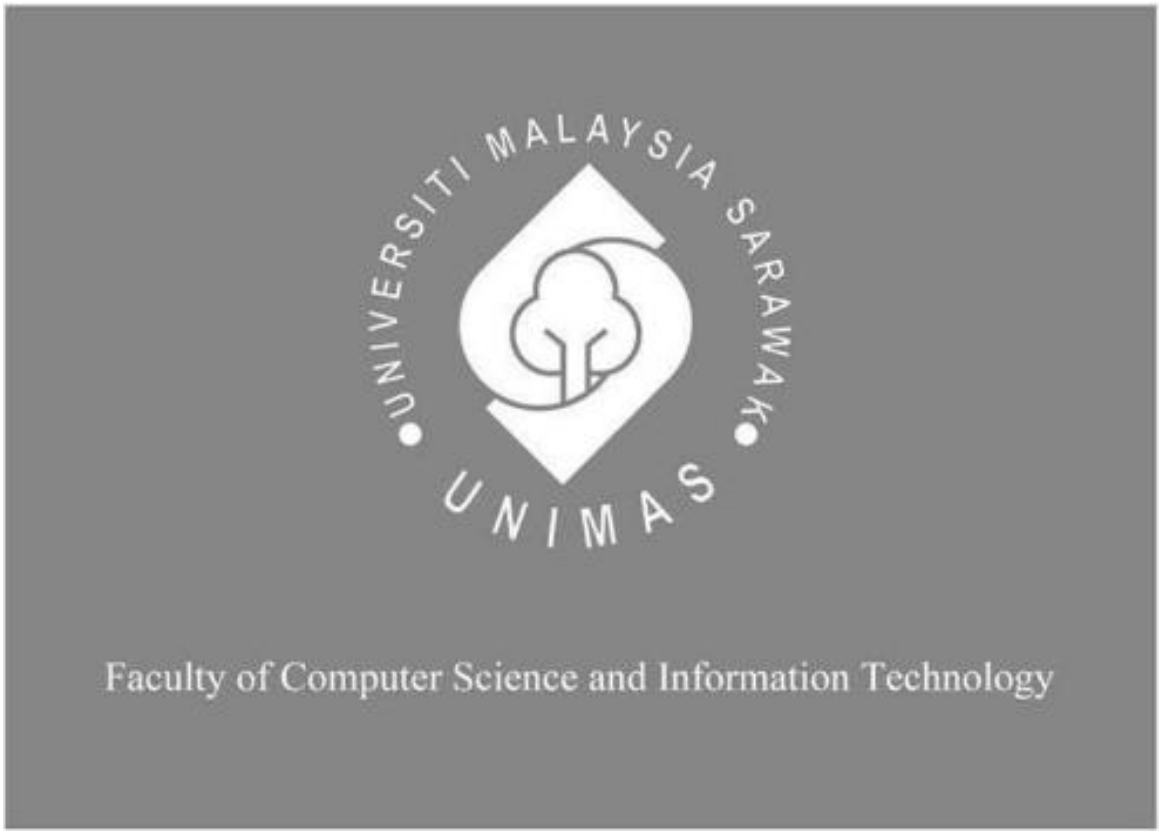


EXTRACTING COLOUR FEATURES FROM COLPOSCOPY IMAGES FOR CLASSIFICATION OF CERVICAL CANCER

Siti Nursyafiqah Binti Abdullah (78541)

Bachelor of Computer Science With Honours

(Computational Science)

2025

EXTRACTING COLOUR FEATURES FROM COLPOSCOPY IMAGES FOR CLASSIFICATION OF CERVICAL CANCER

SITI NURSYAFIQAH BINTI ABDULLAH

This project is submitted in partial fulfilment of
the requirements for the degree of Bachelor of Computer Science with Honours
(Computational Science)

Faculty of Computer Science and Information Technology
UNIVERSITI MALAYSIA SARAWAK
2025

PENGEKSTRAKAN CIRI WARNA DARIPADA IMEJ KOLPOSKOPI UNTUK PENGELASAN KANSER SERVIKS

SITI NURSYAFIQAH BINTI ABDULLAH

Projek ini merupakan salah satu keperluan untuk
Ijazah Sarjana Muda Sains Komputer dengan Kepujian
(Sains Komputan)

Fakulti Sains Komputer dan Teknologi Maklumat
UNIVERSITI MALAYSIA SARAWAK
2025

UNIVERSITI MALAYSIA SARAWAK

THESIS STATUS ENDORSEMENT FORM

TITLE

EXTRACTING COLOUR FEATURES FROM COLPOSCOPY IMAGES FOR CLASSIFICATION OF CERVICAL CANCER

ACADEMIC SESSION: 2024/2025

SITI NURSYAFIQAH BINTI ABDULLAH
(CAPITAL LETTERS)

hereby agree that this Thesis* shall be kept at the Centre for Academic Information Services, Universiti Malaysia Sarawak, subject to the following terms and conditions:

1. The Thesis is solely owned by Universiti Malaysia Sarawak
2. The Centre for Academic Informati on Services is given full rights to produce copies for educational purposes only
3. The Centre for Academic Information Services is given full rights to do digitization in order to develop local content database
4. The Centre for Academic Information Services is given full rights to produce copies of this Thesis as part of its exchange item program between Higher Learning Institutions [or for the purpose of interlibrary loan between HLI]
5. ** Please tick (√)

	CONFIDENTIAL	(Contains classified information bounded by the OFFICIAL SECRETS ACT 1972)
	RESTRICTED	(Contains restricted information as dictated by the body or organization where the research was conducted)
√	UNRESTRICTED	

Validated by

(AUTHOR'S SIGNATURE)

(SUPERVISOR'S SIGNATURE)

Permanent Address

RUMAH KINCHU, BATU 8,
JALAN KJD, SUNGAI MADOR,
96500, BINTANGOR, SARAWAK

Date: 8/7/2025

Date: 8/7/2025

Note * Thesis refers to PhD, Master, and Bachelor Degree
** For Confidential or Restricted materials, please attach relevant documents from relevant organizations / authorities

ACKNOWLEDGEMENT

First and foremost, I would want to express my gratitude to my supervisor, Dr. Stephanie Chua Hui Li, for her outstanding guidance, valuable insights, and constant encouragement throughout this project. Her expertise in data mining and her insightful feedback have greatly influenced my research and assisted me in overcoming challenges. I deeply appreciate her diligence and dedication, which have played an important role in the successful completion of my thesis.

I would also like to extend my deepest thanks to Dr. Wang Yin Chai, our final-year project coordinator, for providing guidelines. I am also grateful to my thesis examiner, Dr Chiew Kang Leng for taking the time to review my work and for providing valuable feedback that has improved the quality of this project. Furthermore, I would like to sincerely thank Faculty of Computer Science and Information Technology (FCSIT) at Universiti Malaysia Sarawak (UNIMAS) for providing me with the opportunity to pursue my studies and for providing the resources that I needed to complete this project.

Besides, I want to thank Roboflow for providing valuable datasets available to me, which were important to my research. My appreciation also goes out to my friends, whose support and guidance have kept me motivated and focused throughout this project.

Finally, I am deeply thankful to my family for their continuous support, motivation, and belief in me, especially my mother which has been my strength throughout this journey.

ABSTRACT

Cervical cancer is a major public health issue that affects millions of women worldwide, highlighting the critical importance of early detection and diagnosis. This project investigates the application of colour feature extraction from colposcopy images to improve the classification accuracy of cervical cancer using machine learning techniques. The research begins with the preprocessing of colposcopy images to improve their quality, which includes resizing all images to a uniform dimension to ensure consistency during colour feature extraction. Colour features are extracted using a variety of methods, including the computation of dominant, mean, and standard deviation values in the Red, Green, Blue (RGB) colour space, enabling a comprehensive analysis of the colour distributions present in the images. The effectiveness to differentiate between normal and abnormal cervical tissues is analysed using machine learning algorithms such as Support Vector Machine (SVM), Random Forest, eXtreme Gradient Boosting (XGBoost), Decision Tree and Logistic Regression. The findings of this study show a significant improvement in classification accuracy, with the extracted colour features providing useful information for the early diagnosis of cervical cancer. Furthermore, the study highlights the potential of automated image processing approaches to help doctors make better diagnostic decisions, ultimately aiming to improve patient outcomes through timely and accurate classification of cervical abnormalities. This study lays a foundation for future developments in automated cervical cancer screening tools, emphasising the integration of colour feature analysis into medical imaging and its value in improving healthcare solutions.

ABSTRAK

Kanser serviks adalah isu kesihatan awam utama yang menjejaskan jutaan wanita di seluruh dunia, menekankan kepentingan kritikal pengesanan dan diagnosis awal. Projek ini menyiasat aplikasi pengekstrakan ciri-ciri warna daripada imej kolposkopi untuk meningkatkan ketepatan pengelasan kanser serviks menggunakan teknik pembelajaran mesin. Penyelidikan ini bermula dengan prapemprosesan imej kolposkopi untuk meningkatkan kualitinya, termasuk penyeragaman saiz imej dengan menukar semua imej kepada dimensi yang sama bagi memastikan konsistensi semasa pengekstrakan ciri warna. Ciri-ciri warna diekstrak menggunakan pelbagai kaedah termasuk pengiraan nilai dominan, min, dan sisihan piawai dalam ruang warna Merah, Hijau, Biru (RGB), yang membolehkan analisis menyeluruh terhadap taburan warna dalam imej. Keberkesanan dalam membezakan antara tisu serviks normal dan tidak normal dianalisis menggunakan algoritma pembelajaran mesin seperti *Support Vector Machine (SVM), Random Forest, eXtreme Gradient Boosting* (*XGBoost), Decision Tree* dan *Logistic Regression.* Penemuan kajian ini menunjukkan peningkatan ketara dalam ketepatan pengelasan, dengan ciri-ciri warna yang diekstrak memberikan maklumat yang berguna untuk diagnosis awal kanser serviks. Tambahan pula, kajian ini menonjolkan potensi pendekatan pemprosesan imej automatik untuk membantu doktor membuat keputusan diagnostik yang lebih baik, dengan matlamat utama untuk meningkatkan hasil pesakit melalui pengelasan abnormaliti serviks yang tepat dan pantas. Kajian ini menjadi asas kepada pembangunan sistem saringan kanser serviks automatik di masa hadapan dengan menekankan integrasi analisis ciri warna dalam pengimejan perubatan dan nilainya dalam meningkatkan penyelesaian penjagaan kesihatan.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Introduction

Cervical cancer is a type of malignant tumour that develops in the cervix which is the lower area of the uterus that connects to the vagina. It develops when abnormal cells in cervix transform into cancerous cells (National Cancer Institute, 2023). These cancerous cells infiltrate the cervix's deeper tissue (stroma) after penetrating the epithelium which known as the outermost layer (National Cancer Society Malaysia, 2018). According to Seng et al. (2020), this condition is often known as “hidden” or “silent” cancer since the patient typically shows no symptoms and the diagnosis is made incidentally during a cervical loop biopsy for a pre-invasive condition. Garland et al. (2008) state that cervical cancer is still a major cause of cancer-related deaths among women globally even though it is largely preventable especially in underdeveloped countries with limited access to healthcare facilities. More than 80% of cancer cases take place in low-income and rural communities, making cervical cancer the most common malignancy among women in these places.

Human Papillomavirus (HPV) Information Centre (2023) reported that in Malaysia, about 991 women lose their lives to cervical cancer each year, while about 1,740 receive a diagnosis. Cervical cancer is the second fastest growing cancer among women between the ages of 15 and 44 and the fourth most common cancer among women nationally. According to Arbyn et al. (2020), the main cause of cervical cancer is a chronic infection with high-risk strains of the human papillomavirus (HPV), which is a common sexually transmitted disease that affects a large percentage of people. It is estimated that about 1.0% of women in the general population are currently infected with high-risk strains of human papillomavirus (HPV) types

16 and 18, which are responsible for 88.7% of invasive cervical cancers (HPV Information Centre, 2023).

According to Cooper and Dunton (2023), colposcopy is an important screening procedure for women's health especially for the detection and prevention of cervical cancer. By magnifying the cervix, vagina, and vulva with a colposcope, this treatment helps doctors spot abnormalities that might not be visible during standard examinations. However, the accuracy of these interpretations can differ significantly depending on the clinician's expertise (Rezende et al., 2021). Researchers are investigating automated image processing methods driven by machine learning to increase consistency and dependability. This method aims to improve the treatment and results for women at risk of cervical cancer by using advanced algorithms to analyse colposcopy images to provide faster and more accurate diagnosis.

The purpose of this project is to investigate the classification of cervical cancer using colour features extracted from colposcopy images. This project applies computational science to develop models to help detect cervical cancer, process, and analyse colposcopy images and extract important colour features. By evaluating the performance of the machine learning algorithms, this study aims to determine the most effective methods for cervical cancer classification and help medical professionals in accurately diagnosing patients. The results of this study might help in the development of automated tools for diagnosis that improve cervical cancer screening's accuracy and efficiency.

1.2 Problem Statement

According to the World Health Organisation (2024), there were over 660,000 new cases of cervical cancer worldwide in 2022, which led to 350,000 deaths. Given that cervical cancer ranks as the second most common cancer among Malaysian women, this statistic indicates the

importance of using efficient screening techniques. However, existing screening technique like Pap (Papanicolaou) smear frequently produce inaccurate results due to limitations in slide processing and cell overlap.

1.3 Objectives

The objectives for this project are:

1. To extract colour features for colposcopy images for the classification of cervical cancer.
2. To evaluate the performance of the machine learning models for classifying cervical cancer
3. To determine the best model for the classification of cervical cancer.

1.4 Methodology

Figure 1.1 below shows the methodologies used for this project.

Figure 1.1: Flowchart for Methodology

This project follows the Cross-Industry Standard Process for Data Mining (CRISP-DM) a well-known and often used framework. It consists of business understanding, data collection, data understanding, data preparation, feature extraction, machine learning, evaluation, and deployment.

1.4.1 Business Understanding

This phase involves identifying the objectives of this project. This is important to provide clear direction to ensure that efforts are aligned toward specific goals.

1.4.2 Data Understanding

The collection of colposcopy image datasets that consist of images of both normal and abnormal cervixes is the primary task that needs to be completed during this phase. The images are sourced from websites like Roboflow, which provides access to relevant datasets on cervical cancer. This dataset includes images divided into two folders: one for positive cases (abnormal) and another for negative cases (normal).

1.4.3 Data Preparation

During this phase, colposcopy images were cropped and resized to a uniform dimension to improve model performance. It also involves data augmentation methods such as rotation and flipping to increase the dataset. Then, the images are labelled as normal or abnormal to prepare them for supervised learning.

1.4.4 Feature extraction

In this process, important features are extracted from the processed images. The extraction of dominant, mean, and standard deviation values across different colour spaces is used to capture the colour distribution, as this project focuses exclusively on colour feature extraction. The collected data provides an in-depth analysis of the images that will be used to train machine learning to classify cervix images by focussing on colour features.

1.4.5 Machine Learning

After feature extraction, traditional machine learning algorithms will be used to classify the images using the extracted colour features.

1.4.6 Evaluation

The performance of the classification model is assessed using key metrics throughout the evaluation stage. This evaluation ensures that the model is dependable and practical for real-world use.

1.4.7 Deployment

In this last phase, the top-performing model is prepared for real-world application. The chosen model will be implemented into a web application for demonstration purposes as part of the project's deployment phase. After being selected for its accuracy and performance, the cervical cancer classification model is uploaded into the web application to allow users to access it.

1.5 Scope

This project's scope includes investigating colour in colposcopy images to improve the classification of cervical cancer by effectively extracting colour features. Its goal is to develop methods which employ the variety of information that colour provides to identify precise details and patterns that are important for an early diagnosis. Colour-based feature extraction will be the focus, with greyscale images purposefully being excluded since they lack the complexity and detail required for accurate abnormality classification. The main goal is to

determine best model for early detection of abnormal cervical tissues, assisting doctors in making accurate clinical assessments.

1.6 Significance of Project

This project is significant for the development of medical knowledge in the field of medicine. It improves professional healthcare's knowledge and skills by integrating machine learning into the classification of cervical cancer. This project encourages the application of the advanced technologies, which enables professionals to sharpen their medical judgement and diagnostic skills. As healthcare professionals becomes more skilled at using these tools, they raise the quality of care they provide while promoting a culture of continuous learning and adaptation in the medical field. This change not only leads to better medical outcomes but also opens doors to new discoveries in cancer care to make sure that patients receive the most effective treatments possible.

Besides, this project greatly benefits communities by making early cervical cancer detection more accessible. It provides access to advanced health care by developing effective and automated diagnostic tools, especially in rural areas with limited medical resources. This can lead to better overall health for everyone and help create a healthier and more resilient community. Furthermore, by developing accessible tools, the project contributes to the reduction of healthcare disparities by ensuring that everyone has access to early screening and treatment despite their economic status or geographical location.

This project also significantly impacts individuals especially women who are at risk of cervical cancer. It makes sure that people can get fast and accurate diagnoses by improving early detection and diagnostic accuracy, which greatly increases the probability that their treatment will be successful. Beyond simply enhancing medical care, this gives patients and

their families hope. In addition, this project allows doctors to offer better patient care by providing personalised and reliable diagnostic methods.

1.7 Expected Outcome

The expected outcome of this project is to evaluate and compare various classification models to identify the most effective and accurate model for distinguishing between normal and abnormal cervical tissues. The goal is to select a model that can reliably classify colposcopy images with high accuracy by refining the image analysis and feature extraction processes. This will ensure the model's suitability for medical use. The best-performing model will serve as a tool to help medical professionals analyse images more efficiently and make well-informed diagnostic decisions.

1.8 Thesis Outline

Chapter 1

Chapter 1 provides an overview of this project. It consists of the problem statement, objectives, methodology, scope, significance of the project, expected outcome, thesis outline, project schedule and summary of Chapter 1.

Chapter 2

Chapter 2 reviews existing research on cervical cancer and the importance of image processing in the analysis of colposcopy images. It will focus on colour feature extraction methods and how machine learning models applied to detect cancer.

Chapter 3

Chapter 3 discusses about the methodology used to solve the problem, focusing on machine learning methods used. It outlines the selected algorithm for the classification model, explain their applicability to the study and compares the performance of machine learning models. The purpose of this chapter is to clearly explain how these methods will be applied to meet the project's goals and provide a detailed comparison of their effectiveness in classifying colposcopy images.

Chapter 4

Chapter 4 outlines the implementation of data mining and data processing procedures, focusing on preprocessing and feature extraction. It describes how machine learning models were trained using the extracted colour features and explains how their performance was assessed. This chapter provides a comprehensive view of the methods used to prepare the data and train the models for image classification.

Chapter 5

Chapter 5 displays the result to show how well the model worked with the extracted colour features. This chapter also discusses about the strength and limitation of colour feature extraction used for classification of cervical cancer. It includes a comparison between machine learning models, evaluating how each approach performs with the colour features. The results are discussed in terms of their effectiveness for cervical cancer classification, highlighting the strengths and weaknesses of each method.

Chapter 6

Chapter 6 is the main conclusion for this project by summarising all of it. It includes contribution to cervical cancer classification, limitations encountered and suggestion for future works.

1.9 Project Schedule

ID	Title	Start Time	End Time
1	Final Year Project 1	10/07/2024	01/25/2025
2	Project Understanding	10/07/2024	11/21/2024
3	Data Collection	10/07/2024	11/21/2024
4	Brief Proposal	10/07/2024	10/20/2024
5	Full Proposal	10/20/2024	11/14/2024
6	Introduction	10/15/2024	11/21/2024
7	Literature Review	10/31/2024	12/13/2024
8	Methodology Planning	11/11/2024	01/05/2025
9	Submission of FYP1 Final Report	01/17/2025	01/17/2025
10	Presentation FYP1	01/24/2025	01/25/2025
11	Final Year Project 2	03/10/2025	07/01/2025
12	Submission of revised FYP report structure	03/10/2025	04/01/2025
13	Data Preprocessing	03/10/2025	03/23/2025
14	Feature Extraction	03/24/2025	05/15/2025
15	Modelling	03/24/2025	05/15/2025
16	Pattern Evaluation	05/15/2025	05/31/2025
17	Deployment	05/15/2025	06/30/2025
18	Reporting	05/31/2025	06/09/2025
19	Submission of Full Report FYP2 & Paper	06/10/2025	06/10/2025
20	Submission of Final Report, Paper & Source Code	06/11/2025	06/23/2025
21	FYP Symposium	06/30/2025	07/01/2025

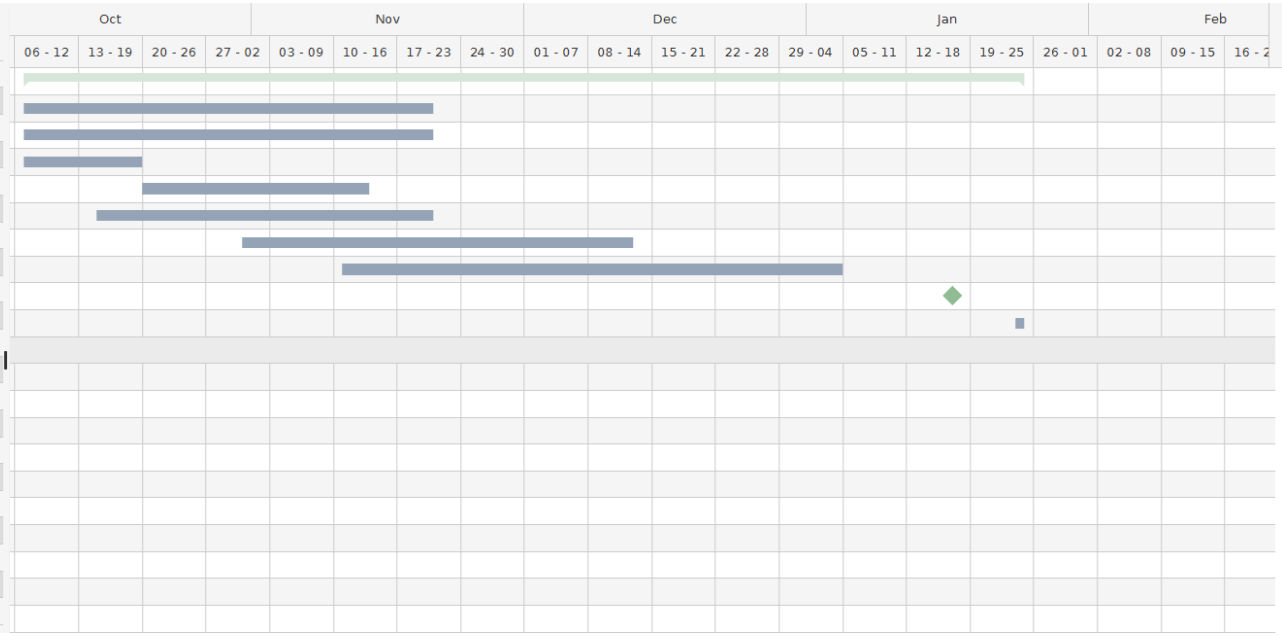
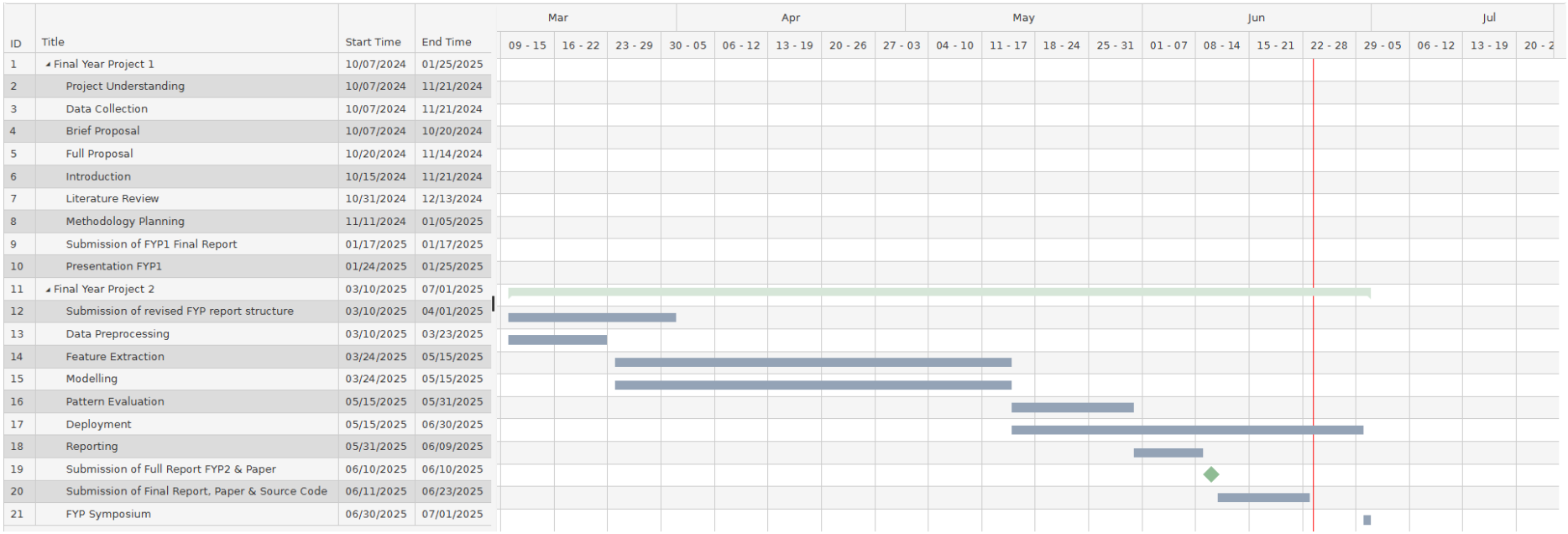


Figure 1.1: Gantt Chart FYP 1



ID	Title	Start Time	End Time
1	Final Year Project 1	10/07/2024	01/25/2025
2	Project Understanding	10/07/2024	11/21/2024
3	Data Collection	10/07/2024	11/21/2024
4	Brief Proposal	10/07/2024	10/20/2024
5	Full Proposal	10/20/2024	11/14/2024
6	Introduction	10/15/2024	11/21/2024
7	Literature Review	10/31/2024	12/13/2024
8	Methodology Planning	11/11/2024	01/05/2025
9	Submission of FYP1 Final Report	01/17/2025	01/17/2025
10	Presentation FYP1	01/24/2025	01/25/2025
11	Final Year Project 2	03/10/2025	07/01/2025
12	Submission of revised FYP report structure	03/10/2025	04/01/2025
13	Data Preprocessing	03/10/2025	03/23/2025
14	Feature Extraction	03/24/2025	05/15/2025
15	Modelling	03/24/2025	05/15/2025
16	Pattern Evaluation	05/15/2025	05/31/2025
17	Deployment	05/15/2025	06/30/2025
18	Reporting	05/31/2025	06/09/2025
19	Submission of Full Report FYP2 & Paper	06/10/2025	06/10/2025
20	Submission of Final Report, Paper & Source Code	06/11/2025	06/23/2025
21	FYP Symposium	06/30/2025	07/01/2025

Figure 1.2: Gantt Chart FYP 2

1.10 Summary

Chapter 1 introduces cervical cancer as a serious health concern especially in rural and low-income areas. It discusses the importance of colposcopy for screening and how automated image processing methods can be used to increase diagnostic accuracy. The project's goal is to improve the classification of cervical cancer by applying a variety of methods to extract colour information from colposcopy images and then developing a reliable classification model. It involves business understanding, data collection, data understanding, data preparation, feature extraction, machine learning, evaluation, and deployment of the most effective model. By using these methodologies, this project seeks to expand knowledge about the cervical cancer among the medical community and increase the accessibility of early detection and treatment for patients.

Chapter 2

Literature Review

2.1 Introduction

This chapter provides a comprehensive review of the existing literature on the application of machine learning in the classification of cervical cancer with a focus on image processing and feature extraction techniques. It explores the use of machine learning for analysing colposcopy images to classify cervical cancer.

In the field of cervical cancer classification, machine learning has significantly changed how doctors analyse and interpret colposcopy images. Machine learning techniques, which rely on predefined characteristics help in classifying cervical tissue by identifying patterns and making predictions based on the data. These technologies reduced human error and the need for manual interpretation as well as improved the accuracy and efficiency of diagnosis.

2.2 Background Knowledge of Cervical Cancer

According to Hull et al. (2020), cervical cancer is a major worldwide health concern especially in low- and middle-income countries (LMICs). The two main types of cancer are adenocarcinoma (AC) and squamous cell carcinoma (SCC). It is up to 90% are categorised as squamous cell carcinoma (SCC) and develops from squamous cell on the outer cervix. Adenocarcinoma (AC), on the other hand, is less common and develops in glandular cells of endocervix. Both types of cancer typically develop in the transformation zone, where the squamous and glandular cells meet, making it a critical area for the formation of precancerous lesions (National Cancer Institute, 2023).

Cervical cancer generally develops over several years in a slow process with several phases. It often starts with cervical intraepithelial neoplasia (CIN), a type of precancerous

change that can sometimes resolve naturally without treatment. However, when high-risk human papillomavirus (HPV) infections persist, cervical intraepithelial neoplasia (CIN) may progress from low-grade to high-grade lesions and lead to serious cervical cancer (Loopik et al., 2020). Human papillomavirus (HPV) is classified as either high-risk (HR) or low-risk depending on its association with precancerous, benign, or cancerous lesions. As the most common and responsible for around 70% of cervical cancer cases globally, high risk subtypes 16 and 18 are also associated with cancers of other organs, including the neck, vulva, and anus (Hull et al., 2020).

Smoking also one of the factors that can increase the cervical cancer as tobacco chemicals may reduce the immune system's defences against human papillomavirus (HPV) infections (Nersesyan et al., 2020). Furthermore, having several sexual partners and using oral contraceptives for a long period of time have been caused to an increased chance of human papillomavirus (HPV) transmission, which in consequence increases the risk of developing cervical cancer (Kashyap et al., 2019).

2.3 Related Works

There are several studies have explored the use of machine learning techniques for the detection and classification of cervical cancer. These methods use a variety of datasets and algorithms, demonstrating improvements in the precision and effectiveness of cervical cancer identification.

Ganguly et al. (2023) used a dataset of 858 patients to predict cervical cancer risk. The preprocessing involved managing missing values, normalising the dataset, and encoding categorical variables. Support Vector Machines (SVM), one of the six machine learning tests, achieved the highest accuracy (99.60%) but the lowest precision (33) and F1 score (0.29). K-

Nearest Neighbours (KNN) achieved a recall of 66 and an accuracy of 97%. Decision Trees (DT) has the highest F1 score (0.92), the best precision (86), and 96% accuracy. XGBoost and Random Forests (RF) both achieved 98% accuracy; RF's F1 scores were like XGBoost's and its recall was 86 while Logistic Regression performed the worst with an F1 score of 0.29. The findings in this paper show that SVM is the most accurate model. Besides, Srivastav et al. (2023) proposed a study using Cervical Cancer Data Set from Kaggle with 859 records and 36 attributes to evaluate Naïve Bayes, Logistic Regression, Decision Tree, and Random Forest. The preprocessing steps included involved converting numeric values to strings, removing duplicates, and handling missing data. The results showed that Decision Tree performed best with an accuracy of 0.968, precision of 0.977, recall of 0.969, and F1-measure of 0.971, followed by Random Forest with an accuracy of 0.961, precision of 0.961, recall of 0.961, and F1-measure of 0.961, Logistic Regression with an accuracy of 0.953, precision of 0.964, recall of 0.953, and F1-measure of 0.957 and the last one is Naïve Bayes with an accuracy of 0.902, precision of 0.950, recall of 0.903, and F1-measure of 0.920.

Furthermore, Rahman and Ghosh (2022) used Cervical Cancer Behaviour Risk Data Set from Kaggle and the UCI machine learning repository, which is consists of 36 features and data from 858 patients. After removing the missing values shown as question marks and converting the data to float format for preprocessing, 668 patients and 34 features were obtained. Six machine learning models were evaluated in this study: Gaussian Naïve Bayes (Accuracy 96.26%, Precision 0.81, Recall 0.85, F1-score 0.83, AUC 0.826), Support Vector Machine (Accuracy 95.52%, Precision 0.85, Recall 0.85, F1-score 0.85, AUC 0.930), Logistic Regression (Accuracy 97.76%, Precision 0.89, Recall 0.83, F1-score 0.86, AUC 0.786), Random Forest (Accuracy 94.77%, Precision 0.97, Recall 0.65, F1-score 0.72, AUC 0.631), K-Nearest Neighbor (Accuracy 91.79%, Precision 0.46, Recall 0.50, F1-score 0.48, AUC 0.500), and Gaussian Naïve Bayes (Accuracy 87.31%, Precision 0.57, Recall 0.92, F1-score

0.57, AUC 0.924). These show that Logistic Regression is the best model in this paper. Rawat et al. (2023) collected data from Kaggle and preprocessing steps included manage missing values and remove duplicates. Then, they trained Logistic Regression, Bagging with Decision Trees, Random Forest with upsampling and class weights, and XGBoost. The Logistic Regression model achieved 95.94% accuracy, 66.70% precision, 72.71% recall, 69.52% F1 score, and an AUC of 0.8242, while the Bagging with Decision Tree model achieved 95.80% accuracy, 70.10% precision, 63.30% recall, and an AUC of 0.9222. The Random Forest with Upsampling and Class Weights model achieved 96.49% accuracy, 72.70% precision, 72.71% recall, and an AUC of 0.8898, and the XGBoost model performed with 97.08% accuracy, 80.11% precision, 72.72% recall, 72.74% F1 score, and an AUC of 0.9586. These results indicate that XGBoost have the best performance compare to other models.

Moreover, Soğukkuyu and Ata (2022) used a dataset from Hospital Universitario de Caracas that contained medical records, demographic information, and behavioural data from 858 patients. They used Recursive Feature Elimination (RFE) for feature selection and the Synthetic Minority Oversampling Technique (SMOTE) technique to deal with missing data and class imbalance. Both Decision Tree and Random Forest models achieved 94% accuracy, but the Decision Tree performed better than Random Forest in recall, achieving 80% recall in predicting cancer patients and 95% recall after applying RFE, while Random Forest had 98% accuracy but only 90% recall. This study found that Decision Tree is the best model for identifying cervical cancer. Sundarambal et al. (2022) also proposed both machine learning approaches to detect cervical cancer early, using data from 858 patients. The data underwent preprocessing to handle missing values, remove irrelevant feature and split into training and testing sets. They tested some machine learning algorithms including Logistic Regression, SVM, Random Forest and Gradient Boosting. SVM performed well, with 97.09% accuracy, 98.78% precision, 98.18% recall, and a 98.47% F-measure. While Gradient Boosting achieved

97.09% accuracy, 98.17% precision, 98.77% recall, and 98.46% F-measure, Random Forest scored 95.93% accuracy, 99.39% precision, 96.44% recall, and 97.89% F-measure. This study proved SVM is the top performed model.

Prasher et al. (2023) used a cervical cancer dataset from Kaggle with 859 records and 36 attributes. Preprocessing involved dealing with missing data, eliminating duplicates, standardising the dataset, and converting "Biopsy" results to categorical labels. They tested four machine learning such as Random Forest (RF), Decision Tree (DT), Multi-Layer Perceptron (MLP), and Logistic Regression (LR) and the process of parameterisation included adjusting hyperparameters such as the splitting criteria (Gini index) for DT and RF and the learning rate for MLP. The results indicated that DT performed extremely well in precision (97.50%), with an accuracy of 96.50% and an F1-score of 0.97, while RF had the highest accuracy and recall (97.00%). LR's F1-score was 0.95, and its accuracy, precision, and recall were all 94.50%. MLP had an F1-score of 0.94 and 94.00% accuracy, precision, and recall. This study concluded that RF is the most effective model as it has the highest accuracy. Besides, Devi et al. (2023) proposed a machine learning approach to predict cervical cancer risk using a Kaggle dataset and ensure that no missing values during data preprocessing. The study applied several classification algorithms, including Logistic Regression (LR), Support Vector Machine (SVM), Decision Tree (DT), and Random Forest (RF), with parameterization involving cross-validation and tuning of hyperparameters like tree numbers, tree depth, and kernel selection. The results showed that Random Forest achieved the highest accuracy at 97.8%, followed by SVM at 97.3%, LR at 96.2% and DT at 96.1%. These findings showed Random Forest is effective at predicting cervical cancer.

In this study, Jha et al. (2021) focused on data manipulation, preprocessing, and machine learning techniques while predicting cervical cancer risk using a Kaggle dataset. Patients were categorised into low-risk and high-risk groups using the processed and visualised

dataset, which included 1492 cases and 37 features. Preprocessing involved dividing the data into training and testing sets, resolving missing values, and performing feature scaling. The study's model used XGBoost, an effective tree-boosting method known for its accuracy and speed. With 97.6% accuracy, 98% precision, 98% recall, and a 98% F1-score after training, the XGBoost model showed excellent accuracy. This study proved that XGBoost can perform better in classifying cervical cancer. In their 2024 study, Uddin et al. employed a dataset of 858 patient samples and 36 attributes from Hospital 'Universitario de Caracas' in Caracas, Venezuela. Preprocessing involved missing numbers, outliers, and label encoding. They used Support Vector Machines (SVM), Random Forest (RF), k-Nearest Neighbors (KNN), Decision Trees (DT), Naive Bayes (NB), Logistic Regression (LR), AdaBoost (AdB), Gradient Boosting (GB), Multilayer Perceptron (MLP), and Nearest Centroid Classifier (NCC). The result showed that the combination of MLP, RF, XGBoost, and PCA achieved an accuracy of 99.19% and 100% sensitivity.

In conclusion, recent studies show that machine learning models such as Support Vector Machines (SVM), k-Nearest Neighbors (k-NN), Decision Tree (DT), Random Forest (RF), Logistic Regression (LR), Gaussian Naïve Bayes (GNB), XGBoost, AdaBoost (AdB), Gradient Boosting (GB), Multilayer Perceptron (MLP), and Nearest Centroid Classifier (NCC) are highly effective in classifying cervical cancer.

2.4 Comparison of related works

Table 2.1 shows the comparison of related works.

Table 2.1: Comparison of all the related works

References	Brief Project Description	Machine Learning Algorithm used	Evaluation metrics	Conclusion
(Ganguly et al., 2023)	This project compares six machine learning algorithms for predicting cervical cancer risk to help to improve early detection and prevention.	• Support Vector Machine (SVM) • K-Nearest Neighbors (KNN) • Decision Tree (DT) • Random Forest (RF) • XGBoost	• Accuracy • Precision • Recall • F1 Score • Receiver Operating Characteristic (ROC)	Support Vector Machine (SVM) model has a best performance in cervical cancer classification compare to other models.

		• Logistic Regression		
(Srivastav et al., 2023)	This project aims to develop a machine learning-based predictive model to accurately classify and identify the presence of cervical cancer.	• Decision Tree (DT) • Random Forest • Naïve Bayes algorithm • Logistic regression	• Accuracy • Precision • Recall • F-measures	Decision Tree is the most effective model as it has the best results in cervical classification.
(Rahman & Ghosh, 2023)	This project used some machine learning models to analyse "Cervical Cancer Behavior Risk Dataset" for early-stage cervical cancer prediction.	• K-Nearest Neighbors (K-NN) • Support Vector Machine (SVM)	• Accuracy • Precision • Recall • F1-Score	Logistic Regression (LR) shows the best performance. In classifying cervical cancer.

		• Decision Tree (DT) • Random Forest (RF) • Logistic Regression (LR) • Gaussian Naïve Bayes (GNB)	• Area Under Curve (AUC)	
(Rawat et al., 2023)	This project explores the use of machine learning algorithms to improve early detection and prediction of cervical cancer.	• Logistic Regression • Bagging with Decision Trees • Random Forest with upsampling and class weights	• Accuracy • Precision • Recall • F1-Score • Area Under Curve (AUC)	This study shows that XGBoost is the best model for predicting cervical cancer as it has the highest accuracy.

		• XGBoost		
(Soğukkuyu & Ata, 2022)	This study investigates machine learning models used to predict the risk of cervical cancer to improve early detection and pre-diagnosis.	• Decision Tree • Random Forest	• Accuracy • Precision • Recall • F1-Score	This study concluded that Decision Tree is the most effective models as it performs better than Random Forest.
(Sundarambal et al., 2022)	By analysing preprocessed data from the UCI repository with machine learning techniques, this study aims to enhance the early detection of cervical cancer and develop more accurate prediction models.	• Logistic Regression • Random Forest • Gradient Boosting • Support vector machine (SVM)	• Accuracy • Precision • Recall • F-measure	This study shows that SVM perform better than other models.
(Prasher et al., 2023)	This project focusses on predicting cervical cancer risk using clinical	• Logistic regression (LR)	• Accuracy • Recall	This study concluded that Random Forest is the most effective

	data and machine learning approaches.	• Multilayer perceptron (MLP) • Decision tree (DT) • Random forest (RF)	• Precision • F1-Score	machine learning algorithms as it outperforms all the other models.
(Devi et al., 2023)	This project focuses on using machine learning techniques to develop an accurate model for predicting cervical cancer to help patients for early detection.	• Decision Tree (DT) • Random Forest (RT) • Support Vector Machine (SVM)	• Accuracy	This study shows that Random Forest (RF) is the best model in classifying cervical cancer with the highest accuracy.
(Jha et al., 2021)	This study shows the effectiveness and dependability of the XGBoost	• XGBoost	• Accuracy • Precision	This study proved that XGBoost performed better that other machine learning algorithms.

	algorithm in predicting the risk of cervical cancer.		• Recall • F1-score	
(Uddin et al., 2024)	This project focusses on investigating cervical cancer risk using machine learning algorithms.	• Support Vector Machines (SVM) • Random Forest (RF) • k-Nearest Neighbors (KNN) • Decision Trees (DT) • Naive Bayes (NB) • Logistic Regression (LR)	• Accuracy • Confusion Matrix • Area Under Curve (AUC)	This study proved that the combination of machine learning algorithms can improve accuracy of cervical cancer classification.

		• AdaBoost (AdB) • Gradient Boosting (GB) • Multilayer Perceptron (MLP) • Nearest Centroid Classifier (NCC)		

Table 2.1 compares recent studies on cervical cancer classification using machine learning (ML) algorithms. Metrics including accuracy, precision, and recall are used to assess some machine learning algorithms such as Support Vector Machines (SVM), Random Forest (RF), Decision Trees (DT), XGBoost, Logistic Regression (LR), Gaussian Naïve Bayes (GNB), AdaBoost (AdB), Gradient Boosting (GB), Multilayer Perceptron (MLP), and Nearest Centroid Classifier (NCC). SVM, RF, and DT often perform best for structured data. Overall, machine learning demonstrates strong performance in classifying cervical cancer datasets.

2.5 Machine Learning Algorithms

Machine learning is a subfield of artificial intelligence (AI) that allows systems and computers to learn and perform better through data analysis without direct task programming. It operates by creating algorithms that find patterns in data and use them for prediction or make decisions based on previously unidentified data. The three primary categories of machine learning are supervised learning, which involves training models on labelled datasets to predict outcomes, unsupervised learning, which looks for hidden patterns or groupings in unlabelled data and reinforcement learning, which instructs systems to make decisions through errors to achieve predetermined objectives (Mahesh, 2018). This project will use five machine learning algorithms such as Support Vector Machine (SVM), Random Forest, XGBoost, Decision Tree and Logistic Regression for cervical cancer classification.

2.5.1 Support Vector Machine (SVM)

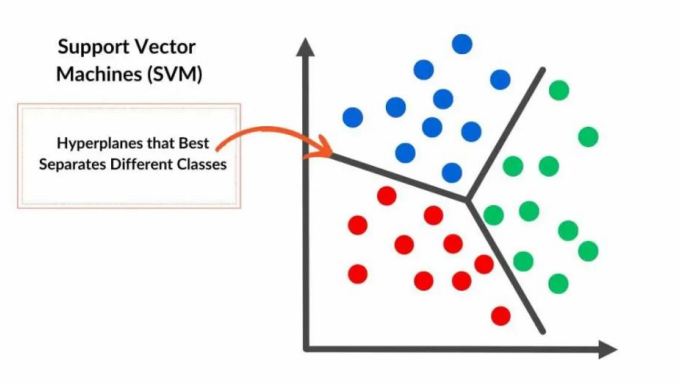


Figure 2.1: The Architecture of Support Vector Machine (Otten, 2024)

Support Vector Machines (SVM) are supervised learning algorithms that are used for problems involving regression and classification. SVM's main goal is to choose the best hyperplane for separating data points from various classes with the greatest margin. It creates a surface or line that separates the data into two groups. In higher dimensions, this hyperplane transforms into a plane or hyperplane, but in two dimensions, it is only a line. SVM seeks to identify the hyperplane that maximises the distance, or margin, between each class's nearest data points. These points which are referred to as support vectors are important for creating the hyperplane. These points are important because they have a direct impact on the hyperplane's orientation and position. The model is not significantly impacted by the other data points, which are located far from the hyperplane. This feature makes SVM especially useful for applications requiring text and image classification that involve huge feature spaces (Cortes & Vapnik, 1995).

However, the classes cannot be accurately divided by a single hyperplane since real-world data is usually not linearly separable. A technique that SVM uses for solving this problem is known as the kernel trick. The data is transformed into a higher-dimensional space using the kernel function, increasing the possibility that it can be separated linearly. This allows the SVM to categorise complex, non-linear data. The linear kernel, polynomial kernel, radial

basis function (RBF) kernel and sigmoid kernel are examples of common kernel functions. SVM is a flexible algorithm that may be used for a variety of tasks, such as medical image analysis and facial recognition as it can handle both linear and non-linear data (Schölkopf et al., 2001).

The SVM algorithm uses optimisation to determine which hyperplane best divides the data. In this optimisation problem, the objective is to minimise classification errors and optimise the margin. It is described as a convex quadratic optimisation problem. This problem is effectively solved using a variety of optimisation techniques, including Sequential Minimal Optimisation (SMO). SVM can classify new data points by determining which side of the hyperplane they fall on once the ideal hyperplane has been identified. It then makes predictions based on this decision boundary (Schölkopf et al., 2001).

In conclusion, SVM is a dependable and efficient machine learning technique, especially for applications involving complicated and huge data. SVM can handle both linear and non-linear classification problems due to the kernel trick, which makes it an effective tool for a variety of real-world applications.

2.5.2 Random Forest

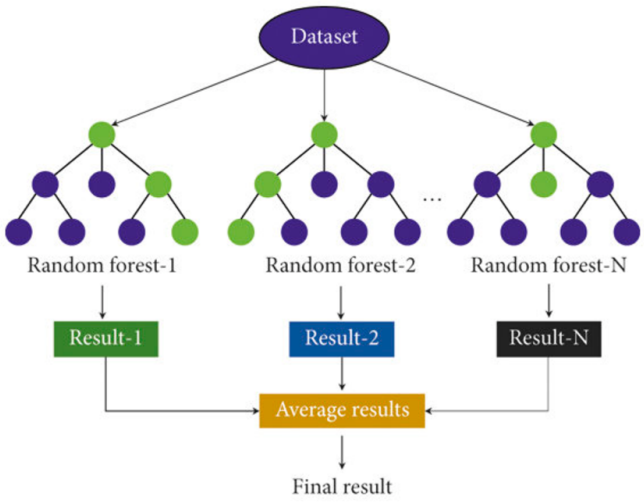


Figure 2.2: The Architecture of Random Forest (Fu & Qi, 2022)

In 2001, Leo Breiman developed Random Forest, an effective method of group learning that combines the results of several decision trees to increase prediction accuracy. Multiple decision tree outputs are used to increase the predictive models' accuracy and adaptability. This approach uses a technique called bootstrapping, where each tree is constructed on a distinct subset of data produced by sampling with replacement. Furthermore, a random subset of features is selected at each node when the data is split, which lowers the correlation between trees. Random Forest has gained widespread adoption in a variety of industries due to its capacity to manage complexity huge data with accuracy (Breiman, 2001).

One of Random Forest's unique features is its Out-of-Bag (OOB) error estimation. The model can be tested using the OOB data, which are data points not included in the bootstrap sample that each tree is trained on. This eliminates the requirement for an independent validation set and offers a fair evaluation of model performance. Furthermore, Random Forest provides a method for determining feature importance that helps in determining which factors have the greatest influence on prediction tasks. This is helpful for huge datasets, where choosing the most important characteristics can improve model performance (Louppe, 2014).

Random Forest's effectiveness depends on optimising its hyperparameters including the minimum number of samples needed for a leaf, the number of trees and the number of features used at each split. Performance can be improved by modifying these parameters as they can balance the relationship between computing efficiency and model complexity. Besides, Random Forest's adaptability in handling a variety of challenges is shown by modifications like Random Survival Forests, which are made for time-to-event data. This makes Random Forest an ideal method for handling complicated and big issues in fields like healthcare due to its versatility and its ability to adapt to noisy data, unbalanced classes, and missing values (Probst el al., 2019).

2.5.3 XGBoost

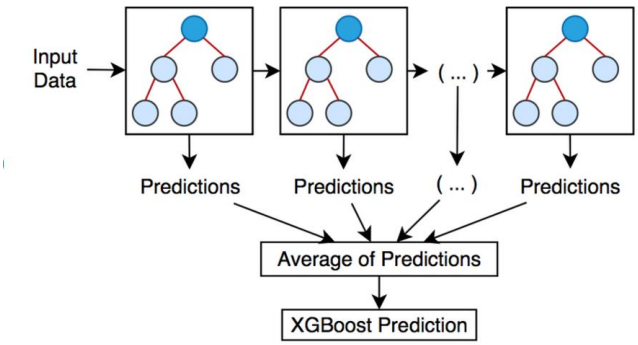


Figure 2.3: The Architecture of XGBoost (Demajo, 2020)

The gradient boosting framework provides the base for the advanced and effective machine learning algorithm known as XGBoost (eXtreme Gradient Boosting). It improves accuracy and generalisation by optimising a regularised objective function using gradient descent and decision trees as weak learners. One of the most effective solutions for dealing with regression and classification problems is XGBoost, which uses decision trees as its base learners and has been adjusted to provide shorter training times and better adaptation. XGBoost

has gained recognition in data science competitions and real-world applications across a variety of industries due to these advancements especially its adaptability and strong performance (Chen & Guestrin, 2016).

XGBoost's architecture is based on gradient-boosted decision trees. The accuracy of the model increases over time by training each new tree to concentrate on the errors (or residuals) of previously trained trees. XGBoost integrates some efficiency-boosting strategies including parallelised tree construction for faster processing, balanced quantile sketch for effective tree splitting and a regularised objective function to reduce overfitting and improve adaptability. The model's scalability and speed are further improved by additional features like learning rate adjustment (shrinkage) and column subsampling. These architectural components ensure XGBoost can handle huge datasets and a variety of learning objectives while being both flexible and precise (S et al., 2016).

XGBoost is known for its accuracy, speed, and adaptability. It is ideal for real-time applications due to its capability to analyse huge datasets quickly and efficiently through the application of distributed and parallel computing. XGBoost is adaptable for several kinds of machine learning problems, from regression to ranking due to its ability to handle missing values and support for flexible objective functions. These advantages strengthen its credibility as a reliable and effective technique for dealing complicated predictive modelling problems in a variety of industries (Dhaliwal et al., 2018).

2.5.4 Decision Tree

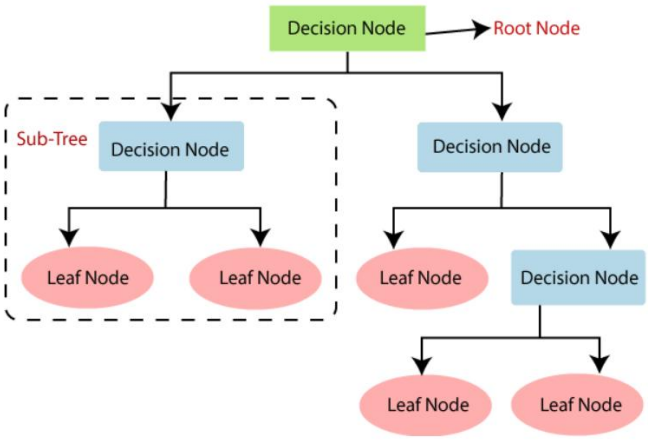


Figure 2.4: The Architecture of Decision Tree (Marios, 2023)

Decision Trees are a widely used machine learning algorithms known for their simplicity, interpretability, and ability to handle both classification and regression tasks. The structure of a decision tree resembles a flowchart, where each internal node represents a test on a feature, each branch corresponds to the outcome of that test, and each leaf node represents a final decision or prediction (Yilmaz, 2023).

The architecture of a decision tree begins at the root node, where the data is initially divided. As the tree grows, internal nodes represent tests on specific features, and branches correspond to outcomes of those tests. The terminal leaf nodes assign a final output or class label. This recursive partitioning continues until a stopping condition is met, such as a maximum depth or minimum number of samples per leaf. Decision trees are prone to overfitting, especially when deep trees are created, but this can be mitigated through pruning techniques or setting constraints on tree growth (Marios, 2023).

Decision trees present key advantages such as ease of visualization, the ability to work without feature scaling, and the capacity to model non-linear patterns in data. However, they are highly sensitive to variations in the training dataset, which can lead to instability in the model's structure. Despite this drawback, decision trees are fundamental to ensemble learning

methods like Random Forests and Gradient Boosting, due to their efficient training process and compatibility with various data types. Their clear structure and rule-based decisions also make them especially valuable in applications where model transparency and interpretability are essential (Yilmaz, 2023).

2.5.3 Logistic Regression

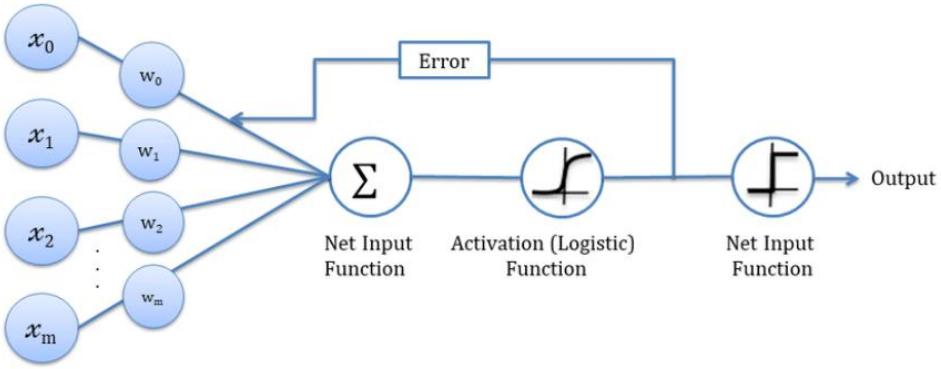


Figure 2.5: The Architecture of Logistic Regression (Nobel et al., 2024)

Logistic Regression is a commonly used machine learning algorithm that is particularly effective for binary classification tasks. It predicts the probability of an instance belonging to a particular class by modelling the relationship between input features and the target variable through a logistic (sigmoid) function. This function maps any real-valued number into a range between 0 and 1, allowing the model to express the likelihood of outcomes in probabilistic terms. The method is grounded in statistical theory and provides a clear, interpretable output, making it a frequent choice for applications where transparency and explanation of results are essential (Nobel et al., 2024)

One of the primary advantages of logistic regression lies in its simplicity and computational efficiency. Unlike more complex models, it does not require extensive computational resources and performs well on linearly separable datasets. Additionally, the

model's coefficients can be interpreted as the influence of individual features on the predicted outcome, which is especially valuable in fields such as healthcare and social sciences. Logistic regression is also relatively robust to irrelevant features and multicollinearity, especially when regularization techniques like L1 (Lasso) or L2 (Ridge) are applied to enhance generalisation (Hosmer et al., 2013).

However, logistic regression assumes a linear relationship between the independent variables and the log-odds of the dependent variable, which may not hold true for more complex datasets. As a result, the model may underperform when non-linear patterns are present in the data. Despite this, logistic regression continues to serve as a strong baseline model due to its interpretability, ease of implementation, and solid performance in a wide range of real-world applications (Hosmer et al., 2013).

2.6 Evaluation metrics

This phase involves application of three key evaluation metrics: Accuracy, F1 Score and Area Under the Curve (AUC), which are used to evaluate and compare model performance in this project.

2.6.1 Accuracy

Accuracy is one of the most common evaluation metrics used in classification problems. It represents the percentage of accurate predictions (including true positives and true negatives) that the model produced overall (Grandini et al., 2020).

Formula:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Where:

TP: True Positives, TN: True Negatives, FP: False Positives, FN: False Negatives (Grandini et al., 2020).

While accuracy can provide an overview of a model's performance, it can be unreliable with imbalanced datasets. In certain situations, a model may obtain a high accuracy score by only favouring the majority class, which may result in overly biased towards predicting non-cancerous cases (Kulkarni, 2021).

2.6.2 F1-Score

The F1-Score is a metric that balances both Precision (the accuracy of positive predictions) and Recall (the ability to identify all positive cases), making it especially useful in situations where both false positives and false negatives are important (Grandini et al., 2020).

Formula:

$$F1 = 2\left(\frac{Precision \times Recall}{Precision + Recall}\right)$$

Where:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

This formula provides a comprehensive measure of model performance by ensuring that the accuracy of those positive predictions (Precision) and the ability to correctly identify positive cases (Recall) are taken into equal consideration (Grandini et al., 2020). This is

especially important in situations like medical analysis when decisions and patient care may be significantly impacted by both false positives and false negatives.

2.6.3 Area Under Curve (AUC)

The Area Under the Curve (AUC) is a performance metric commonly used to evaluate classification models, especially in binary tasks. It is derived from the Receiver Operating Characteristic (ROC) curve, which plots the True Positive Rate (TPR) against the False Positive Rate (FPR) for different classification thresholds. The AUC offers a comprehensive assessment of a model's ability to differentiate between positive and negative cases at every threshold, with higher values indicating better performance (Kulkarni et al., 2021). AUC is very useful in medical diagnostics since it summarises a model's overall accuracy independent of the threshold used for classification, which makes it suitable for scenarios like cancer detection where the costs of false positives and false negatives must be balanced.

The AUC is calculated as the integral of the True Positive Rate (TPR) over the False Positive Rate (FPR):

$$AUC = \int_0^1 TPR(FPR)dFPR$$

Where:

$$TPR = \frac{TP}{TP + FN}$$

Where TP is the number of false positives and FN is the number of false negatives

$$FPR = \frac{FP}{FP + TN}$$

Where FP is the number of false positives and TN is the number of true negatives.

The range of AUC is from 0 to 1. Perfect discrimination is represented by an AUC of 1, whereas random guessing is indicated by an AUC of 0.5 (Kulkarni et al., 2021). This makes AUC a reliable and comprehensive metric, especially for medical field where it is important to accurately identify true conditions while avoiding unnecessary medical treatments.

2.7 Tools

The tools used in this project are Python and its key libraries, which provided the foundation for implementing machine learning models. Python was chosen because it is flexible and has an extensive framework that includes reliable libraries to simplify tasks in data processing, and model evaluation. These tools supported the effective execution of the classification task.

Scikit-learn was used for providing a systematic method to model evaluation by performing data preparation tasks like normalisation, splitting, and assessing classification metrics like accuracy and F1-score (Shanmugamani, 2018).

Several important libraries were used in this project to help with different process stages. While OpenCV was used for image preparation, making sure the data was accurately formatted and compatible with the machine learning models, Pandas and NumPy were important for managing data and performing numerical operations (Miller, 2018). Matplotlib was used to create plots like accuracy graphs and confusion matrices that helped explain the results and effectively visualise the model's performance (Rougier, 2021).

2.8 Summary

Chapter 2 discusses about the background of cervical cancer where it focusses on cervical cancer development and its factors. This chapter also reviews some previous research papers that related to this project especially machine learning algorithms used for cervical cancer classification. A table of comparison of previous works also included in this chapter to highlight the methodologies, evaluation metrics and performance outcomes of these studies. Besides, this chapter also introduces some machine learning algorithms will be used in this project, highlighting their architectures and advantages.

Chapter 3

Methodology

3.1 Introduction

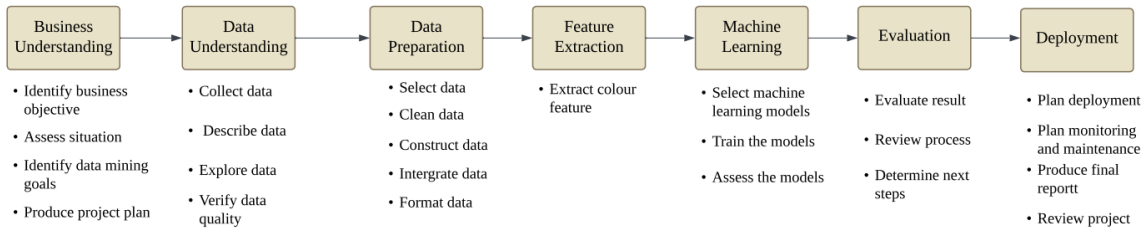


Figure 3.1: Flowchart of Proposed Methodology

This chapter provides a detailed explanation of the methodology used in this project. Figure 3.1 shows a flowchart of the proposed methodology, outlining the phases from defining business objectives to deploying the final model. The phases include Business Understanding, Data Understanding, Data Preparation, Feature Extraction, Machine Learning, Evaluation and Deployment. Each phase is important to provide an effective and reliable model to classify cervical cancer using colposcopy images.

3.2 Business Understanding

This phase lays the groundwork for the project by defining its primary aims and objectives. The main goal is to use colposcopy images to develop a reliable tool for early cervical cancer classification. In areas with limited resources, early detection is important for improving patient outcomes and lowering mortality rates. This project involves extracting colour features from colposcopy images for machine learning classification. This project involves extracting colour features from colposcopy images for machine learning classification. The models are evaluated to identify the most effective one.

The primary challenges include the complexity of medical imaging, inconsistent image quality and the requirement for accurate classification. Therefore, the study focusses on extracting colour-based features to improve the efficacy and reliability of classification algorithms to overcome these challenges. A detailed project plan has been created, outlining the necessary resources and methodologies to effectively overcome these challenges and achieve the project's objectives.

3.3 Data Understanding

The dataset used in this study consists of a total of 2,617 colposcopy images, which are sourced from online platforms such as Roboflow. The images are divided into two categories which are positive cases (abnormal) with 1172 colposcopy images and negative cases (normal) with 1445 colposcopy images (VIA, 2024).

The dataset performed some preprocessing steps to ensure it was good quality and suitable for analysis. According to VIA (2024), the images' alignment was adjusted using Auto-Orient to ensure each image is properly positioned and upright. This step minimises any problems that might arise from images being flipped or rotated, which could impair the model's accuracy in interpreting them. After that, Static Crop was used to highlight the most important parts of the images by retaining only the middle 20–80% of the vertical and horizontal areas. As a result, the model may concentrate on the important traits related to cervical cancer detection by removing redundant borders and background noise. Lastly, every image was enlarged to 640×640 pixels, which is a consistent resolution.

3.4 Data Preparation

Refining the dataset for analysis is the main goal of the data preparation step. The initial stage is to choose relevant data from the collected dataset based on predetermined standards, like image clarity and resolution. After that, the data is cleaned to remove any problems including noise, outliers, or missing values to make sure consistency throughout the dataset.

The process of improving the dataset to make it more suitable for analysis is known as data construction. This includes modifying existing ones to represent the data more accurately. Since multiple datasets are being used, data integration is used to combine them into one, standard format. Once the data is standardised, it is ready for use with machine learning algorithms. As part of this preparation, colposcopy images are resized to 224 × 224 pixels, a standard dimension commonly used to ensure consistency and compatibility with machine learning models. These steps are important for constructing a high-quality dataset that improves the performance and accuracy of the models.

3.5 Feature Extraction

The next step involves extracting colour features from the colposcopy images. This project focuses on colour-based feature extraction using both statistical and perceptual methods, including the calculation of the mean and standard deviation of Red, Green and Blue (RGB) values, as well as the identification of the dominant colour (RGB) in each image.

3.6 Machine Learning

In this project, Support Vector Machine (SVM), Random Forest, Extreme Gradient Boosting (XGBoost), Decision Tree and Logistic Regression will be used to classify

colposcopy images into normal and abnormal classes based on colour-based features extracted from RGB mean, standard deviation and colour dominant.

For the SVM model, the focus is on finding the optimal hyperplane to split the two classes effectively. The Radial Basis Function (RBF) kernel is chosen for its ability to represent non-linear interactions that are often present in real-world datasets. Besides, some parameters like gamma, which determines the impact of individual data points on the model, and the regularisation parameter (C), which manages the trade-off between training accuracy and model complexity will be adjusted to obtain the best balance between unfitting and overfitting. The model will also be trained using normalised colour features for data consistency to maximise the performance and increase the accuracy.

Next, Random Forest algorithm, which decreases overfitting and increases dependability by developing a group of decision trees will also be used. The model will be trained using decision trees to provide a varied set of models. Besides, parameters such as maximum depth of the trees and the number of features considered for each split will be adjusted. This method provides a reliable classification process for the dataset.

Besides, XGBoost, known for its effectiveness and speed will be used. The model will be trained with boosting rounds, adding a new tree each time, and the learning rate to control the effect of each new tree on the final prediction to minimise overfitting. The maximum tree depth will be adjusted to reduce each tree's complexity. Like the other models, XGBoost will also be trained using colour features for data consistency to improve the performance.

The Decision Tree algorithm will also be used to classify colposcopy images based on the extracted colour features. This model works by splitting the dataset into subsets using feature values that result in the highest information gain or Gini index reduction. The decision paths formed create a tree structure that leads to final classification outcomes. Parameters such

as the maximum depth of the tree, minimum samples per leaf will be fine-tuned to balance model complexity and performance. The model also will be trained on extracted colour features to improve classification performance.

Lastly, Logistic Regression which is a fundamental classification algorithm will be used for cervical cancer classification. This algorithm applies a sigmoid function to map the input features to a probability value between 0 and 1, which is then used to classify images as normal or abnormal. Key parameters such as the regularisation strength (C), solver, and maximum number of iterations will be adjusted to prevent overfitting and ensure the model generalises well to unseen data. Despite its simplicity, Logistic Regression can serve as a strong baseline and often yields competitive results when combined with well-selected features.

Each of these models will be trained and modified to ensure they are suitable for classifying colposcopy images. Using different types of machine learning models helps to identify different patterns and structures in the data, which can improve classification accuracy.

3.7 Evaluation

This project will use three evaluation metrics such as accuracy, F1-Score and AUC to evaluate the performance of models. Since the dataset is balanced, accuracy is an effective metric to assess overall performance of the models as it measures the percentage of correct predictions to the total number of predictions made. However, F1-Score will also be used to obtain more detailed evaluation of the model because F1-Score provides a balance between precision (the number of predicted positive case that are identified correctly) and recall (the number of actual positive cases that are correctly predicted). Therefore, the F1-score is important in preventing overfitting to either class by maintaining a balance between minimising errors. Lastly, AUC is calculated using Receiver Operating Characteristic (ROC)

curve to evaluate the models' ability to differentiate between the two classes (normal and abnormal) at different threshold levels. A higher AUC shows that the model is effectively classify between the two classes, regardless threshold level has been set.

3.8 Deployment

HuggingFace will be used to deploy the top-performing selected model for cervical cancer classification based on the performance results to ensure the model achieves the required accuracy and reliability. The model will be exported in a format compatible with HuggingFace and uploaded to both Model Hub and Hugging Face Spaces for easy access and interaction. Hugging Face Spaces will host an interactive web application using Gradio, allowing users to upload colposcopy images and receive real-time predictions through a user-friendly interface. Additionally, through API requests, HuggingFace's RESTful API provides real-time predictions. The deployed system in this project will use colposcopy images as input and makes predictions to classify the images as normal or abnormal. Hugging Face also provides tools for monitoring and managing the deployed models to make sure they keep operating effectively and are accessible.

3.9 Summary

This chapter describes the methodology for classifying cervical cancer using colposcopy images. The method starts with defining the goals of the project and understanding the dataset, which consists of 2,617 images that have been categorised as normal and abnormal. The preprocessing step, like resizing and standardising, is also included in this chapter. Colour-based features will be extracted in RGB colour space for machine learning models such as Support Vector Machine (SVM), Random Forest, XGBoost, Decision Tree and Logistic

Regression to increase classification accuracy. The accuracy, F1-Score, and AUC of the models will be used to evaluate their precision and reliability. Finally, the trained models will be deployed by HuggingFace for easy accessibility for real-world application.

Chapter 4

Implementation

4.1 Introduction

This chapter presents the implementation phase of the project, which focuses on transforming the proposed methodology into a functional tool for cervical cancer classification using colposcopy images.

The implementation is carried out using the Python programming language due to its simplicity, flexibility, and extensive library support for image processing and machine learning. The entire process includes preparing the colposcopy image dataset, applying preprocessing techniques, extracting colour features, training classification models, and evaluating their performance. The implementation focuses on extracting RGB-based features such as mean, standard deviation, and dominant colour, which are then used to train models like Support Vector Machine (SVM), Random Forest, XGBoost, Decision Tree, and Logistic Regression. The best performing model will be selected for deployment.

4.2 Implementation

The implementation of the cervical cancer classification follows a systematic approach, from data preparation and preprocessing to feature extraction, classification, and performance evaluation. Each of these steps are important in ensuring that the system can accurately distinguish between normal and abnormal cervix images based on colour features. The implementation process was carried out using Python and a variety of libraries that are used for image processing, machine learning, and data analysis. Key libraries used in this project include:

- OpenCV: Used for image processing tasks such as cropping, resizing, and converting images to different colour spaces (RGB).
- NumPy: Used for numerical operations on image arrays, including calculations of mean and standard deviation of pixel values.
- Pandas: Used for organising and manipulating the dataset, storing extracted features, and managing labels during the model training and testing phases.
- scikit-learn: Used for building machine learning models, splitting datasets, and evaluating model performance using metrics such as accuracy, precision, recall, and F1-score.
- KMeans (from scikit-learn): Used to extract the dominant colour from images by clustering RGB values.
- XGBoost: An efficient and scalable implementation of gradient boosting used for training classification models.
- Matplotlib and Seaborn: These libraries were used to visualise model performance through plots such as ROC curves to help in the interpretation of results.
- LabelEncoder: Used to convert categorical labels into numerical format for machine learning model compatibility.
- Joblib: Enabled saving of trained models for later use and deployment to ensure the reproducibility of results.

Each of these libraries are important in ensuring a smooth and efficient implementation of the cervical cancer classification system. The entire implementation process was conducted using the Visual Studio Code (VS Code) environment, which offered a flexible and organised platform for writing, testing, and debugging Python code throughout the project.

4.2.1 Data Preparation

Resize

Figure 4.1: Raw image size 640x480 pixels

The original colposcopy images, as illustrated in Figure 4.1, varied in size and resolution, typically around 640×480 pixels. This inconsistency in dimensions could potentially introduce bias and hinder the performance of machine learning algorithms, which require uniform input for optimal training and evaluation.

Figure 4.2: Resized image to 224x224 pixels

To address this, a preprocessing step was implemented to resize all images to a standard dimension of 224×224 pixels, as shown in Figure 4.2. This resizing ensured uniformity across the dataset, reduced computational overhead, and aligned the data structure with the input

requirements of most machine learning models. The resizing was done using the OpenCV library, which enabled efficient handling of image transformations.

The resizing process was automated using a Python script that iterated through the dataset, resized each image, and saved the output to a designated folder. Both normal and abnormal cervix images were processed using the same procedure to maintain uniformity and fairness during the model training phase. By standardising the image sizes, the dataset was optimally prepared for the next phases of feature extraction and classification.

Split the data

```python
def split_data(original_dir, train_dir, val_dir, test_dir, test_size=0.15, val_size=0.15):

    class_dirs = ['Normal', 'Abnormal']

    for class_dir in class_dirs:
        class_dir_path = os.path.join(original_dir, class_dir)


        if not os.path.exists(class_dir_path) or not os.listdir(class_dir_path):
            print(f"Warning: The directory {class_dir_path} is empty or does not exist. Skipping...")
            continue


        class_files = [f for f in os.listdir(class_dir_path) if os.path.isfile(os.path.join(class_dir_path, f))]


        train_files, temp_files = train_test_split(class_files, test_size=0.3, random_state=42)


        val_files, test_files = train_test_split(temp_files, test_size=0.5, random_state=42)


        os.makedirs(os.path.join(train_dir, class_dir), exist_ok=True)
        os.makedirs(os.path.join(val_dir, class_dir), exist_ok=True)
        os.makedirs(os.path.join(test_dir, class_dir), exist_ok=True)
```

Figure 4.3: Python Code for Splitting the Dataset into Training, Validation, and Testing Sets

After resizing, the dataset was split into three subsets with 70% used for training, 15% for validation, and 15% for testing, as shown in Figure 4.3. Each subset was organised and saved into separate folders named train, validation, and test, respectively.

4.2.2 Feature Extraction

```python
def get_color_stats(image_path):
    image = cv2.imread(image_path)
    if image is None:
        return None

    image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)

    r, g, b = cv2.split(image)

    r_mean, g_mean, b_mean = np.mean(r), np.mean(g), np.mean(b)
    r_std, g_std, b_std = np.std(r), np.std(g), np.std(b)

    pixels = image.reshape(-1, 3)
    kmeans = KMeans(n_clusters=1, random_state=42, n_init=10)
    kmeans.fit(pixels)
    dominant_color = kmeans.cluster_centers_[0]
    r_dom, g_dom, b_dom = dominant_color

    return {

        'R_mean': float(r_mean), 'G_mean': float(g_mean), 'B_mean': float(b_mean),
        'R_std': float(r_std), 'G_std': float(g_std), 'B_std': float(b_std),
        'R_dom': float(r_dom), 'G_dom': float(g_dom), 'B_dom': float(b_dom),
    }
```

Figure 4.4: Python function to extract RGB mean, standard deviation, and dominant colour from colposcopy images

OpenCV and scikit-learn were used to create a Python script that extracted colour-based features from the colposcopy images. This function uses KMeans clustering to determine the dominant colour as well as the mean and standard deviation of the red (R), green (G), and blue (B) colour channels. The colour that is most visible in the image is known as the dominant colour, and it provides insight into the overall colour composition. A separate function was used to iterate through all images in both the Normal and Abnormal folders, apply the colour statistics extraction, and compile the results into a structured format. The code of the feature extraction process is illustrated in Figure 4.4.

	A	B	C	D	E	F	G	H	I	J	K
1	R_mean	G_mean	B_mean	R_std	G_std	B_std	R_dom	G_dom	B_dom	Image	Label
2	225.7013	159.8274	180.4348	39.02872	54.92235	53.95934	225.7013	159.8274	180.4348	AAA1_flipp	0
3	206.1934	137.6984	151.6899	48.32211	64.87444	67.31765	206.1934	137.6984	151.6899	AAA1_nois	0
4	221.3969	158.6831	179.0233	37.69998	53.86169	53.08732	221.3969	158.6831	179.0233	AAA1_nois	0
5	221.3616	158.7463	179.0137	37.61619	53.85827	52.97203	221.3616	158.7463	179.0137	AAA1_nois	0
6	221.4093	158.6961	178.9352	37.63603	53.82675	52.9843	221.4093	158.6961	178.9352	AAA1_nois	0
7	206.1548	137.7874	151.7035	48.28555	64.89386	67.17782	206.1548	137.7874	151.7035	AAA1_nois	0
8	221.3475	158.6873	178.9185	37.61677	53.85904	52.94266	221.3475	158.6873	178.9185	AAA1_nois	0
9	209.2897	138.2283	152.5235	51.636	67.0658	69.4225	209.2897	138.2283	152.5235	AAA1_rota	0
10	208.8483	138.7318	153.0915	53.23134	67.1692	69.34648	208.8483	138.7318	153.0915	AAA1_rota	0
11	226.017	159.7949	180.4491	38.49921	54.32442	53.02347	226.017	159.7949	180.4491	AAA1_rota	0
12	226.0076	159.8334	180.5671	38.47864	54.2633	52.94678	226.0076	159.8334	180.5671	AAA1_rota	0
13	197.7612	127.5055	120.3754	56.70154	39.51469	41.16133	197.7612	127.5055	120.3754	AAG1_flipp	0
14	197.6648	127.4237	120.2736	56.86566	39.68803	41.29433	197.6648	127.4237	120.2736	AAG1_jpg.	0
15	194.4507	126.883	119.6956	55.05745	39.76918	41.64684	194.4507	126.883	119.6956	AAG1_nois	0
16	182.327	126.3986	120.6031	54.19173	51.15704	51.35081	182.327	126.3986	120.6031	AAG1_nois	0
17	182.2988	126.3551	120.5891	54.20286	51.12723	51.39343	182.2988	126.3551	120.5891	AAG1_nois	0
18	182.3378	126.4083	120.6466	54.22225	51.09991	51.35387	182.3378	126.4083	120.6466	AAG1_nois	0
19	194.5893	126.9155	119.7699	55.05785	39.67408	41.59953	194.5893	126.9155	119.7699	AAG1_nois	0
20	194.5278	126.8597	119.7232	55.01265	39.76577	41.56096	194.5278	126.8597	119.7232	AAG1_nois	0

Figure 4.5: First 20 sample of extracted features from dataset

The final dataset, as illustrated in Figure 4.5, comprises the image filenames, extracted statistical colour features, and corresponding class labels (0 for Normal and 1 for Abnormal). This structured dataset is exported to a CSV file and subsequently used as input for further analysis and model training.

4.2.3 Machine Learning

Support Vector Machine (SVM)

```
param_grid = {
    'C': [0.1, 1, 10, 100],
    'gamma': [0.01, 0.1, 1, 10],
    'kernel': ['rbf']
}

svm = SVC(probability=True, class_weight='balanced')

grid_search = GridSearchCV(svm, param_grid, cv=5, scoring='accuracy')
grid_search.fit(X_train, y_train)

best_model = grid_search.best_estimator_

best_model.fit(X_train, y_train)
```

Figure 4.6: Support Vector Machine for Cervical Cancer Classification

The Support Vector Machine (SVM) model for cervical cancer classification was developed and optimised using the GridSearchCV function from scikit-learn, which performs exhaustive search over specified hyperparameter values. As shown in Figure 4.6, the model was trained using grid_search.fit(X_train, y_train) where grid_search was defined with an SVM estimator and a parameter grid containing key hyperparameters. Before training, the input features were standardised using StandardScaler to ensure that all features contributed equally to the model's decision boundary. This scaling is particularly important for SVM, which is sensitive to the relative scales of input features. This function not only trains the model but also applies 5-fold cross-validation to evaluate its performance across multiple data splits, helping to prevent overfitting.

The hyperparameter grid explored combinations of C values [0.1, 1, 10, 100], which regulate the penalty for misclassified samples, and gamma values [0.01, 0.1, 1, 10], which control the influence of each training sample on the decision boundary. The kernel used was 'rbf' (Radial Basis Function), which is suitable for handling non-linear relationships in the input data. These settings allow the model to learn a flexible and accurate classification boundary between normal and abnormal cervical images.

During training, GridSearchCV fits multiple SVM models using different combinations of hyperparameters and identifies the best-performing configuration based on cross-validated scores. Once the best combination is found, the SVM model is automatically retrained on the full training set using these optimal hyperparameters. The final model is then extracted using grid_search.best_estimator_ and saved via joblib.dump(best_model, 'svm_color_model.pkl'). This ensures that the trained model is both optimally tuned and readily reusable for future predictions.

Random Forest

```
param_grid = {
    'n_estimators': [100, 200, 300],
    'max_depth': [10, 20, 30, None],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4]
}

rf = RandomForestClassifier(random_state=42, class_weight='balanced', bootstrap=False)

grid_search = GridSearchCV(rf, param_grid, cv=5, n_jobs=-1, verbose=0)
grid_search.fit(X_train, y_train)

best_model = grid_search.best_estimator_

best_model.fit(X_train, y_train)
```

Figure 4.7: Random Forest for Cervical Cancer Classification

The Random Forest (RF) model for cervical cancer classification was constructed and fine-tuned using the GridSearchCV function from scikit-learn. As illustrated in Figure 4.7, the model was trained using grid_search.fit(X_train, y_train) where grid_search was defined with a Random Forest classifier and a hyperparameter grid. Before model training, the input features were standardised using StandardScaler to maintain consistency in data preprocessing across

all models. The parameter grid included variations of n_estimators [100, 200, 300], max_depth [10, 20, 30, None], min_samples_split [2, 5, 10] and min_samples_leaf [1,2,4]. Through 5-fold cross-validation, the training process ensured a more generalised model by evaluating performance across different data splits and reducing the likelihood of overfitting.

During training, GridSearchCV evaluates multiple Random Forest models with different hyperparameter combinations and identifies the best-performing configuration based on cross-validation scores. Once the optimal set of parameters is determined, the model is automatically retrained on the full training dataset. The final best model is then retrieved using grid_search.best_estimator_ and saved via joblib.dump(best_model, 'rf_model.pkl'), making it both performance-optimized and reusable for future cervical cancer predictions.

XGBoost

```
param_grid = {
    'n_estimators': [100, 200, 300],
    'learning_rate': [0.01, 0.1, 0.2],
    'max_depth': [3, 6, 10]
}

xgb = XGBClassifier(use_label_encoder=False, eval_metric='logloss', random_state=42)
grid_search = GridSearchCV(xgb, param_grid, cv=5, n_jobs=-1, verbose=0)
grid_search.fit(X_train, y_train)

best_model = grid_search.best_estimator_
best_model.fit(X_train, y_train)
```

Figure 4.8: XGBoost for Cervical Cancer Classification

The XGBoost model used for classifying cervical images was developed and optimised using the GridSearchCV method, which exhaustively explores combinations of hyperparameters to improve performance. As shown in Figure 4.8, training was performed with grid_search.fit(X_train, y_train), using a parameter grid that tested n_estimators [100, 200, 300], learning_rate [0.01, 0.1, 0.2], and max_depth [3, 6, 10]. Before training, the feature values were standardised using StandardScaler to ensure that all input features contributed equally to the model's learning process. This process used 5-fold cross-validation to ensure robust performance and reduce overfitting by evaluating the model across multiple data partitions.

During the grid search, various XGBoost configurations were trained and validated, and the one with the highest cross-validation score was selected. After identifying the best hyperparameters, the XGBoost model was automatically retrained on the full training set to finalise learning. The best estimator was then accessed using grid_search.best_estimator_ and saved using joblib.dump(best_model, 'xgb_color_model.pkl'), ensuring the model is ready for accurate and repeatable future use.

Decision Tree

```
param_grid = {
    'max_depth': [10, 20, 30, None],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4]
}

dt = DecisionTreeClassifier(random_state=42, class_weight='balanced')
grid_search = GridSearchCV(dt, param_grid, cv=5, n_jobs=-1, verbose=0)
grid_search.fit(X_train, y_train)

best_model = grid_search.best_estimator_
best_model.fit(X_train, y_train)
```

Figure 4.9: Decision Tree for Cervical Cancer Classification

The Decision Tree classifier for cervical cancer detection was built and fine-tuned using GridSearchCV, enabling systematic hyperparameter optimisation. As shown in Figure 4.9, the model was trained using grid_search.fit(X_train, y_train), where grid_search was defined with a Decision Tree classifier and a hyperparameter grid tailored to control the tree's growth and complexity. The grid included four values for max_depth [10, 20, 30, None], which defines the maximum depth of the tree, three values for min_samples_split [2, 5, 10], which sets the minimum number of samples required to split an internal node, and three values for min_samples_leaf [1, 2, 4], which determines the minimum number of samples required to be at a leaf node. Prior to model training, the feature values were standardised using StandardScaler to ensure consistent scaling across all input features, supporting stable and reliable model learning

GridSearchCV evaluated multiple Decision Tree models based on the defined hyperparameters and selected the configuration that yielded the highest cross-validation score. Once the optimal hyperparameters were identified, the Decision Tree was retrained using the full training data. The best model was retrieved with grid_search.best_estimator_ and saved using joblib.dump(best_model, 'decision_tree_color_model.pkl'), enabling efficient reuse for cervical cancer classification tasks.

Logistic Regression

```
param_grid = {
    'C': [0.1, 1, 10, 100],
    'solver': ['liblinear', 'lbfgs'],
    'max_iter': [100, 200, 300]
}

log_reg = LogisticRegression(class_weight='balanced')
grid_search = GridSearchCV(log_reg, param_grid, cv=5)
grid_search.fit(X_train, y_train)

best_model = grid_search.best_estimator_

best_model.fit(X_train, y_train)
```

Figure 4.10: Logistic Regression for Cervical Cancer Classification

The Logistic Regression model for cervical cancer classification was optimised using GridSearchCV to explore different hyperparameters. As shown in Figure 4.10, the model was trained using the grid_search.fit(X_train, y_train) function, where grid_search was configured with a Logistic Regression estimator and a hyperparameter grid. The grid included several regularisation strengths C [0.1, 1, 10, 100], two solvers, 'liblinear' and 'lbfgs', and three possible maximum iteration values for optimisation [100, 200, 300]. The 5-fold cross-validation method was used to evaluate the model's performance on multiple subsets of the training data, reducing the risk of overfitting and improving generalisation. Before training, all feature values were standardised using StandardScaler from scikit-learn. This preprocessing step scaled the input data to have zero mean and unit variance, ensuring that all features contributed equally to the model training and reducing potential bias due to differing feature scales.

GridSearchCV trained multiple Logistic Regression models under different hyperparameter settings and selected the one with the best cross-validated accuracy. Following this, the model was retrained on the complete training dataset using the best-found parameters. The optimised model was then extracted using grid_search.best_estimator_ and saved using joblib.dump(best_model, 'logistic_regression_color_model.pkl'), allowing it to be reused for future predictions with improved confidence and performance.

4.2.4 Evaluation

```
# --- Evaluate on training set ---
y_train_pred = best_model.predict(X_train)
y_train_proba = best_model.predict_proba(X_train)[:, 1]

train_roc_auc = roc_auc_score(y_train, y_train_proba)
train_fpr, train_tpr, _ = roc_curve(y_train, y_train_proba)

print("Training set evaluation:")
print(f"Accuracy: {accuracy_score(y_train, y_train_pred):.4f}")
print(f"F1-score: {f1_score(y_train, y_train_pred):.4f}")
print(f"ROC AUC: {train_roc_auc:.4f}")
print(classification_report(y_train, y_train_pred, target_names=['Normal', 'Abnormal']))

plt.figure(figsize=(8,6))
plt.plot(train_fpr, train_tpr, label=f'Training ROC Curve (AUC = {train_roc_auc:.2f})')
plt.plot([0, 1], [0, 1], 'k--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Training ROC Curve')
plt.legend()
plt.show()
```

Figure 4.11: Evaluation training set

After training the Support Vector Machine (SVM) model using GridSearchCV, an evaluation was performed on the training dataset to examine how well the model had learned from the data. As illustrated in Figure 4.11, the .predict() function was used to generate predicted labels for the training data. The model's performance was then assessed using accuracy_score() and classification_report() from the sklearn.metrics module, which provided detailed metrics such as precision, recall, and F1-score for each class (Normal and Abnormal). Additionally, the ROC curve was plotted using roc_curve() based on the predicted probabilities obtained from predict_proba(), and the Area Under the Curve (AUC) was computed using roc_auc_score() to evaluate the model's ability to distinguish between the two classes.

```python
# --- Evaluate on test set ---
y_test_pred = best_model.predict(X_test)
y_test_proba = best_model.predict_proba(X_test)[:, 1]

test_roc_auc = roc_auc_score(y_test, y_test_proba)
test_fpr, test_tpr, _ = roc_curve(y_test, y_test_proba)

print("Test set evaluation:")
print(f"Accuracy: {accuracy_score(y_test, y_test_pred):.4f}")
print(f"F1-score: {f1_score(y_test, y_test_pred):.4f}")
print(f"ROC AUC: {test_roc_auc:.4f}")
print(classification_report(y_test, y_test_pred, target_names=['Normal', 'Abnormal']))

plt.figure(figsize=(8,6))
plt.plot(test_fpr, test_tpr, label=f'Test ROC Curve (AUC = {test_roc_auc:.2f})')
plt.plot([0, 1], [0, 1], 'k--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Test ROC Curve')
plt.legend()
plt.show()
```

Figure 4.12: Evaluation on test set

The final evaluation of the model was conducted on the test dataset using a similar process. As illustrated in Figure 4.12 the .predict() method was applied to generate class predictions, and key performance metrics such as accuracy, precision, recall, and F1-score were computed using accuracy_score() and classification_report() from the sklearn.metrics module. Additionally, the predicted probabilities obtained using the .predict_proba() method, were used to plot the Receiver Operating Characteristic (ROC) curve using roc_curve(), while the Area Under the Curve (AUC) was calculated using roc_auc_score() to to assess the model's classification effectiveness.

The same evaluation process was consistently applied to other machine learning models, including Random Forest, Decision Tree, Logistic Regression, and XGBoost. For each model, predictions on the test set were made using .predict(), followed by performance evaluation using accuracy, classification metrics, and ROC-AUC analysis. ROC curves were plotted for visual comparison, and AUC values were calculated based on predicted probabilities to provide a quantitative performance measure.

All these evaluation steps were implemented in Python using libraries such as scikit-learn, matplotlib, and seaborn, and the trained model was saved for future use with joblib.dump().

4.2.5 Deployment

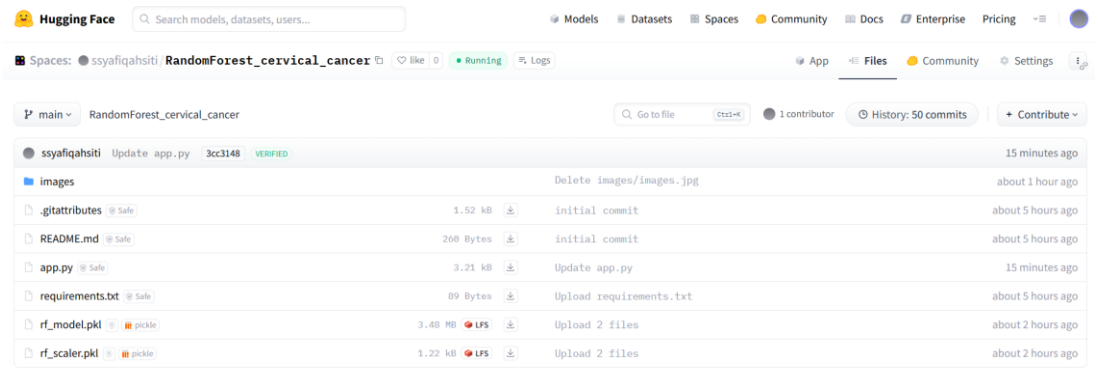


Figure 4.13: HuggingFace Space Interface

The Random Forest model for cervical cancer classification was deployed on HuggingFace Spaces, a platform for hosting machine learning models with a user-friendly web interface. After logging in, a new Space was created to host the application, using Gradio for the front-end. This allowed users to interact with the model by inputting data and receiving real-time predictions. The deployment involved uploading four key files which are the trained ‘rf_model.pkl’, ‘rf_scaler.pkl’, the ‘app.py’ script that created the web interface, and the ‘requirements.txt’ file listing necessary dependencies as shown in Figure 4.13. Additionally, a folder named “images” was included to store example images for user reference. Once uploaded, the Space was launched, generating a public link that allowed users to access the application directly through their browsers.

```
with gr.Blocks(css="""
.centered-container {
    max-width: 600px;
    margin: 0 auto;
    padding: 20px;
}
""") as demo:
    with gr.Column(elem_classes="centered-container"):
        gr.Markdown("##  RandomForest Colour Classifier for Colposcopy Images")
        gr.Markdown("Upload a colposcopy image to classify it as **Normal** or **Abnormal** using RGB colour features.")

        image_input = gr.Image(type="numpy", label="  Upload Colposcopy Image")
        output_text = gr.Textbox(label="  Prediction & Confidence")

        image_input.change(fn=predict_rf_color_model, inputs=image_input, outputs=output_text)

        gr.Markdown("### 🔍 Example Images")
        with gr.Row():
            example_normal_button = gr.Button("Show Normal Example Image")
            example_abnormal_button = gr.Button("Show Abnormal Example Image")

        example_image = gr.Image(label="  Reference Example Only")

        example_normal_button.click(fn=show_example_image_normal, outputs=example_image)
        example_abnormal_button.click(fn=show_example_image_abnormal, outputs=example_image)

# Launch app
if __name__ == "__main__":
    demo.launch()
```

Figure 4.14: Code Snippet from 'app.py'

The ‘app.py’ script used Gradio to create a simple interface where users could input colposcopy images and receive classification predictions (Normal or Abnormal). This interactive interface made the model accessible for real-time testing, and predictions were displayed immediately after user input. Figure 4.14 presents a code snippet from the ‘app.py’ script, illustrating the core Gradio implementation that connects the model to the user interface.

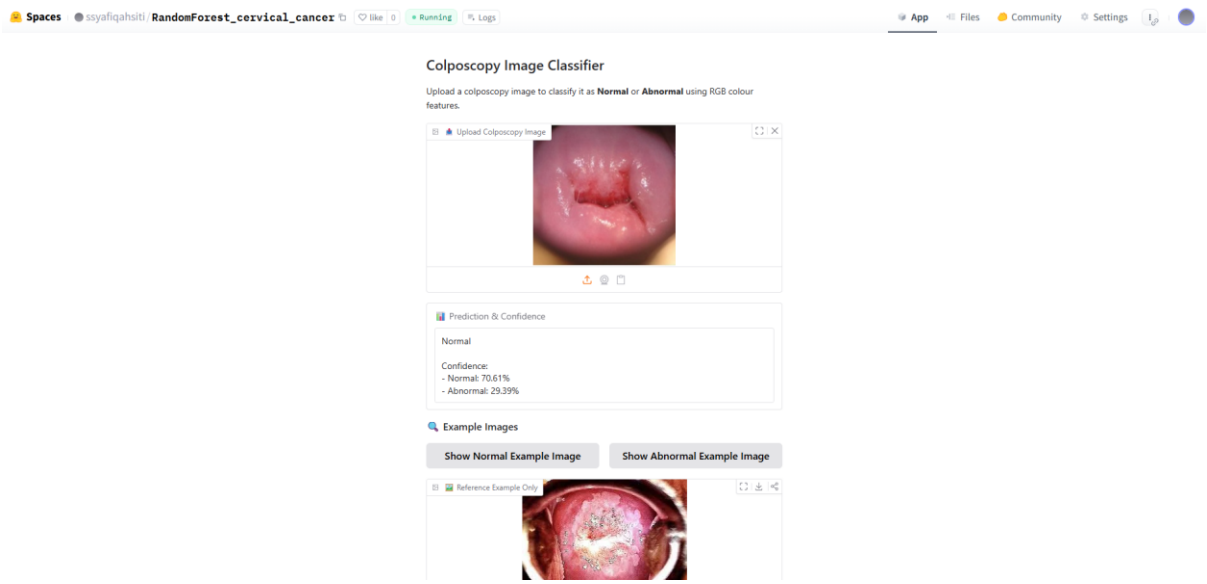


Figure 4.15: Prediction Result

Once the model and application were deployed, users could test the model’s functionality through the interface, providing valuable insights into the performance of the cervical cancer classifier. Figure 4.15 displays the Gradio interface showing the prediction results (‘Normal’ or ‘Abnormal’) after user input. The interface also presents the confidence percentage for the prediction and includes a button that allows users to view example images for reference.

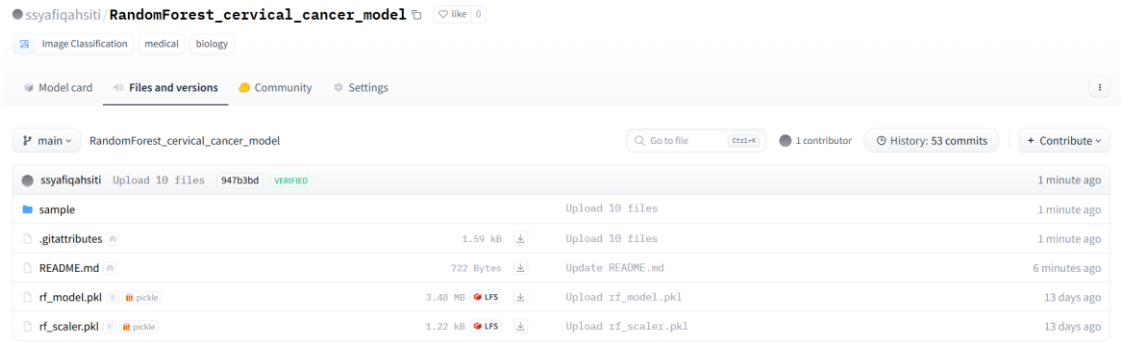


Figure 4.16: Uploaded Random Forest model file on Hugging Face Model Hub

In addition to deployment via Hugging Face Spaces, the trained model (rf_model.pkl) and scaler (rf_scaler.pkl) were uploaded to the Hugging Face Model Hub, as shown in Figure 4.16. This platform offers a centralized, version-controlled environment for efficient model management and sharing. The application script (app.py) loads the model directly from the repository, ensuring seamless integration and consistent preprocessing during inference. The repository also features a sample_images folder containing example colposcopy images, allowing users to test and observe the model's classification performance.

4.3 Summary

This chapter described the implementation process of the cervical cancer classification system using colposcopy images. Python was used due to its strong support for image processing and machine learning. The images were preprocessed by resizing them to 224 × 224 pixels and subsequently split into training, validation, and testing subsets. Feature extraction was performed by calculating RGB-based statistical features, including the mean, standard deviation, and dominant colour. The extracted features were then standardised using StandardScaler. Five machine learning models including SVM, Random Forest, XGBoost, Decision Tree, and Logistic Regression were trained using GridSearchCV for hyperparameter

tuning. The performance of these models was evaluated using accuracy, AUC-ROC curve, and F1 score to determine the most effective approach for classifying cervical images.

Chapter 5

Result and Discussion

5.1 Introduction

This chapter presents the results and discussion of the classification performance based on colour features extracted from cervical images. Five machine learning models including Support Vector Machine (SVM), Random Forest, XGBoost, Decision Tree, Logistic Regression were applied to evaluate their effectiveness in distinguishing between normal and abnormal cervical conditions. The models were assessed using accuracy, ROC curve, and F1 score to provide a balanced view of performance. These evaluation metrics help determine each model's capability in handling classification tasks, especially in terms of precision, recall, and overall reliability. The discussion also highlights key findings, compares model performances, and reflects on possible factors influencing the results.

5.2 Result for Support Vector Machine (SVM)

```
Training set evaluation:
Accuracy: 0.7141
F1-score: 0.6990
ROC AUC: 0.7915
              precision    recall  f1-score   support

      Normal       0.77      0.69      0.73      1006
    Abnormal       0.66      0.74      0.70       816

    accuracy                           0.71      1822
   macro avg       0.71      0.72      0.71      1822
weighted avg       0.72      0.71      0.71      1822
```

Figure 5.1: Performance of SVM Training Set

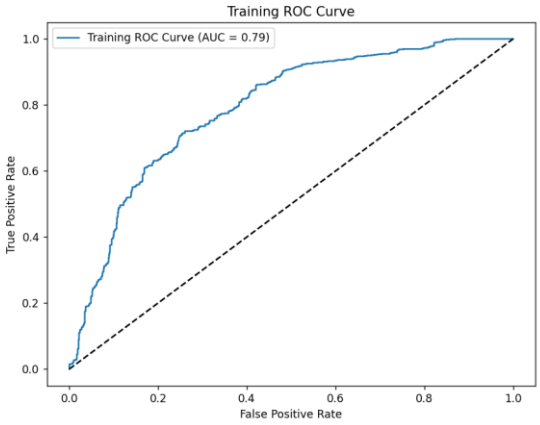


Figure 5.2: ROC Curve for SVM Training Set

The Support Vector Machine (SVM) model achieved an overall accuracy of 71.41% on the training set, indicating that it correctly classified most colposcopy images. The F1-score of 69.90% reflects a balanced trade-off between precision and recall, which is important in medical diagnostics to reduce both false positives and false negatives. For the Normal class, the model attained a precision of 77% and a recall of 695, resulting in an F1-score of 73%, demonstrating effective identification of normal cases with reasonable accuracy. In comparison, the Abnormal class achieved a precision of 66% and a higher recall of 74%, indicating that the model detected abnormal cases with greater sensitivity but slightly lower precision. The ROC-AUC score 79.15% further highlights the model's ability to distinguish between normal and abnormal classes. Overall, these results indicate that the SVM successfully captured meaningful colour-based features from the colposcopy images, though additional refinement may improve classification performance. The detailed performance metrics are

presented in Figure 5.1, while Figure 5.2 illustrates the ROC curve demonstrating the model's discriminative capability.

```
Test set evaluation:
Accuracy: 0.7194
F1-score: 0.6978
ROC AUC: 0.7884
              precision    recall  f1-score   support

      Normal       0.76      0.72      0.74       216
    Abnormal       0.68      0.72      0.70       176

    accuracy                           0.72       392
   macro avg       0.72      0.72      0.72       392
weighted avg       0.72      0.72      0.72       392

Best hyperparameters found: {'C': 1, 'gamma': 0.01, 'kernel': 'rbf'}
```

Figure 5.3: Performance SVM on Test Set

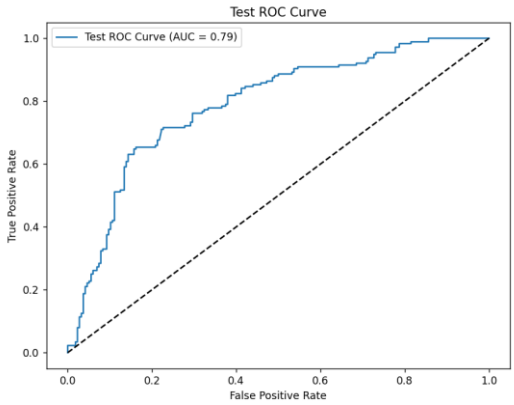


Figure 5.4: ROC Curve for SVM Test Set

The Support Vector Machine (SVM) model achieved an accuracy of 71.94% on the test set, demonstrating reliable classification of unseen colposcopy images. The F1-score of 69.78% indicates a balanced performance between precision and recall. For the Normal class, precision and recall were 76% and 72%, respectively, with an F1-score of 74%. The Abnormal class showed a precision of 68% and recall of 72%, resulting in an F1-score of 70%. The ROC-AUC score of 78.84% confirms the model's effective discrimination between classes. Macro and weighted averages for precision, recall, and F1-score were consistently 72%. The best hyperparameters found were C = 1, gamma = 0.01, and an RBF kernel. These results,

summarised in Figure 5.3 and Figure 5.4, illustrate the model's strong generalisation capability on new data.

5.3 Result for Random Forest

```
Training set evaluation:
Accuracy: 0.9868
F1-score: 0.9854
ROC AUC: 0.9995
              precision    recall  f1-score   support

      Normal       1.00      0.98      0.99      1006
    Abnormal       0.98      1.00      0.99       816

    accuracy                           0.99      1822
   macro avg       0.99      0.99      0.99      1822
weighted avg       0.99      0.99      0.99      1822
```

Figure 5.5: Performance of Random Forest on Training set

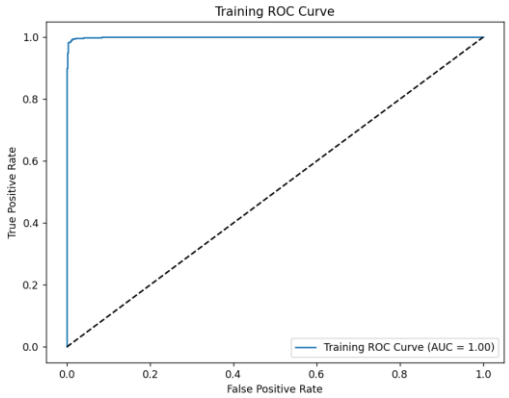


Figure 5.6: ROC Curve for Random Forest Training Set

The Random Forest (RF) model demonstrated excellent performance on the training dataset, achieving an overall accuracy of 98.68%, an F1-score of 98.54%, and a ROC AUC of 99.95%. For the Normal class, the model attained perfect precision of 100% and a recall of 98%, resulting in an F1-score of 99%. Similarly, the Abnormal class showed a precision of 98% and a perfect recall of 100%, with an F1-score of 99%. These results indicate that the model effectively learned to distinguish between normal and abnormal colposcopy images with high confidence. The aggregated metrics further confirm this strong performance, with both macro and weighted averages reaching 99%. Detailed metrics are presented in Figure 5.5, while

Figure 5.6 illustrates the ROC curve reflecting the model's ability to differentiate between classes.

```
Test set evaluation:
Accuracy: 0.9617
F1-score: 0.9584
ROC AUC: 0.9946
              precision    recall  f1-score   support

      Normal       0.99      0.94      0.96       216
    Abnormal       0.94      0.98      0.96       176

    accuracy                           0.96       392
   macro avg       0.96      0.96      0.96       392
weighted avg       0.96      0.96      0.96       392

Best hyperparameters found: {'max_depth': 10, 'min_samples_leaf': 4, 'min_samples_split': 2, 'n_estimators': 200}
```

Figure 5.7: Performance of Random Forest on Test Set

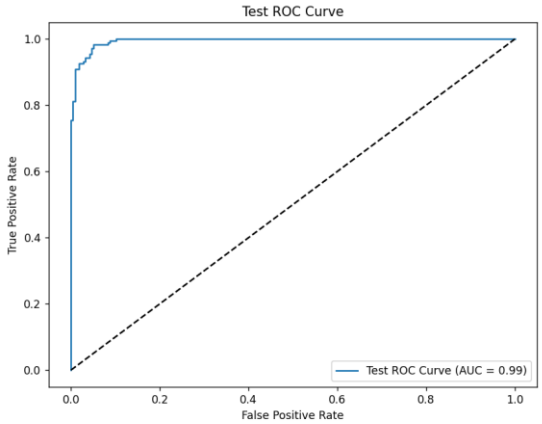


Figure 5.8: ROC Curve for Random Forest on Test Set

The Random Forest model demonstrated strong performance on the test set, achieving an accuracy of 96.17%, an F1-score of 95.84%, and an ROC AUC of 99.46%. For the Normal class, the model achieved a precision of 99% and recall of 94%, resulting in an F1-score of 96%. In contrast, the Abnormal class attained a precision of 94% and recall of 98%, also obtaining an F1-score of 96%. These metrics reflect the model's balanced ability to correctly identify both classes with high confidence. The optimal hyperparameters identified were a maximum depth of 10, minimum samples per leaf of 4, minimum samples split of 2, and 200 estimators. The detailed classification report is shown in Figure 5.7, while Figure 5.8 presents

the ROC curve, illustrating the model's strong capacity to distinguish between Normal and Abnormal cases.

5.4 Result for XGBoost

```
Training set evaluation:
Accuracy: 0.7651
F1-score: 0.7214
ROC AUC: 0.8537
              precision    recall  f1-score   support

      Normal       0.76      0.83      0.80      1006
    Abnormal       0.77      0.68      0.72       816

    accuracy                           0.77      1822
   macro avg       0.77      0.76      0.76      1822
weighted avg       0.77      0.77      0.76      1822
```

Figure 5.9: Performance of XGBoost on Training Set

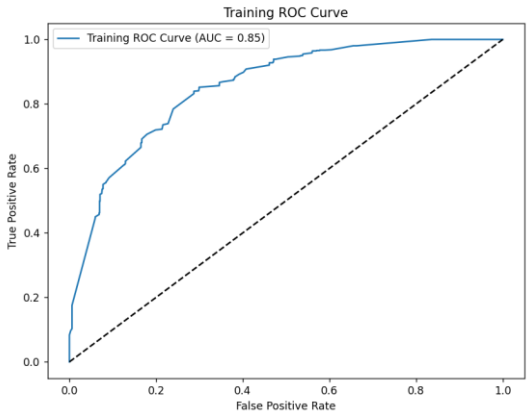


Figure 5.10: ROC Curve of XGBoost on Training Set

The XGBoost model achieved an overall accuracy of 76.51%, with an F1-score of 72.14% and an ROC AUC of 85.37% on the training dataset. For the Normal class, the model demonstrated a precision of 76% and a recall of 83%, resulting in an F1-score of 80%, indicating strong performance in identifying normal cases. The Abnormal class showed a precision of 77% and a recall of 68%, yielding an F1-score of 72%, which reflects good detection capability but slightly lower sensitivity. The macro and weighted averages for precision, recall, and F1-score ranged from 76% to 77%, indicating consistent classification

performance across both classes. These results suggest that the XGBoost model effectively learned relevant patterns from the training data. The detailed performance metrics are presented in Figure 5.9, while Figure 5.10 illustrates the ROC curve highlighting the model's classification capability.

```
Test set evaluation:
Accuracy: 0.7500
F1-score: 0.7066
ROC AUC: 0.8336
              precision    recall  f1-score   support

      Normal       0.75      0.81      0.78       216
    Abnormal       0.75      0.67      0.71       176

    accuracy                           0.75       392
   macro avg       0.75      0.74      0.74       392
weighted avg       0.75      0.75      0.75       392

Best Parameters: {'learning_rate': 0.01, 'max_depth': 3, 'n_estimators': 100}
```

Figure 5.11: Performance of XGBoost on Test Set

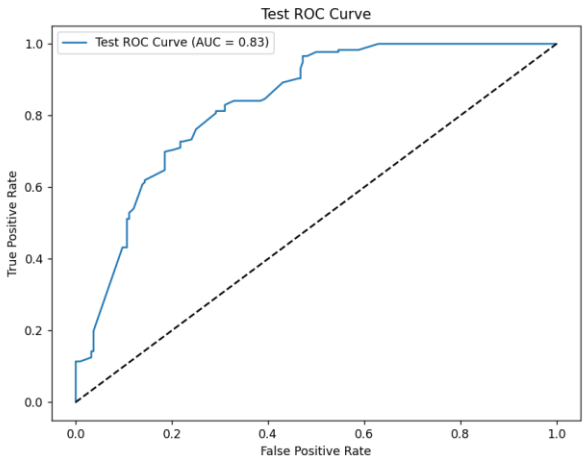


Figure 5.12: ROC Curve of XGBoost on Test Set

The XGBoost model's evaluation on the test set achieved an overall accuracy of 75%, an F1-score of 70.66%, and a ROC AUC of 83.36%, indicating moderate performance in distinguishing between Normal and Abnormal classes. Class-wise metrics revealed balanced precision of 75% for both classes, with recall values of 81% for Normal and 67% for Abnormal cases, reflecting slightly higher sensitivity toward the Normal class. Accordingly, the F1-score

was higher for Normal (78%) than for Abnormal (71%). Macro and weighted averages for precision, recall, and F1-score ranged from approximately 0.74 to 0.75, suggesting consistent overall classification despite class imbalance. The best hyperparameters identified included a learning rate of 0.01, a maximum tree depth of 3, and 100 estimators. These results are presented in Figure 5.11, with the detailed classification report and the ROC curve shown in Figure 5.12, illustrating the model's performance.

5.5 Result for Decision Tree

```
Training set evaluation:
Accuracy: 0.9226
F1-score: 0.9155
ROC AUC: 0.9838
              precision    recall  f1-score   support

      Normal       0.95      0.91      0.93      1006
    Abnormal       0.90      0.94      0.92       816

    accuracy                           0.92      1822
   macro avg       0.92      0.92      0.92      1822
weighted avg       0.92      0.92      0.92      1822
```

Figure 5.13: Performance of Decision Tree on Training Set

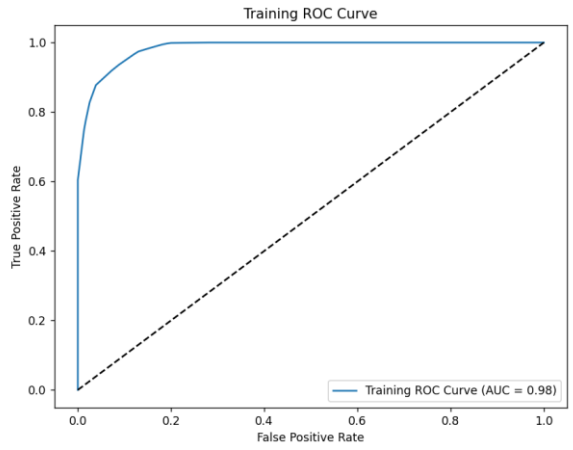


Figure 5.14: ROC Curve of Decision Tree on Training Set

The Decision Tree model's performance on the training set indicates a high level of classification accuracy, achieving 92.26%. Its F1-score of 91.55% reflects a well-balanced

trade-off between precision and recall. The model's ROC-AUC score of 98.38% highlights its strong ability to differentiate in separating Normal from Abnormal cases. Precision values of 95% for Normal and 90% for Abnormal demonstrate the model's reliability in correctly predicting each class, while recall rates of 91% and 94% for Normal and Abnormal respectively indicate its effectiveness in capturing actual positive instances. The F1-scores of 0.93 and 0.92 for the Normal and Abnormal classes further illustrate consistent predictive performance. Additionally, both the macro and weighted averages of precision, recall, and F1-score hover around 0.92, suggesting balanced results across the different categories despite any class imbalances present in the training data. These results are shown in Figure 5.13 and Figure 5.14.

```
Test set evaluation:
Accuracy: 0.8776
F1-score: 0.8667
ROC AUC: 0.9453
              precision    recall  f1-score   support

      Normal       0.90      0.87      0.89       216
    Abnormal       0.85      0.89      0.87       176

    accuracy                           0.88       392
   macro avg       0.88      0.88      0.88       392
weighted avg       0.88      0.88      0.88       392

Best hyperparameters found: {'max depth': 10, 'min samples leaf': 4, 'min samples split': 2}
```

Figure 5.15: Performance of Decision Tree on Test Set

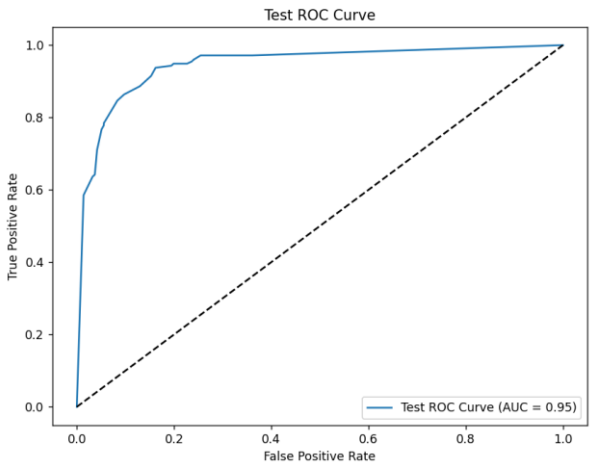


Figure 5.16: ROC Curve of Decision Tree on Test Set

The Decision Tree model achieved an accuracy of 87.76% on the test set, with an F1-score of 86.67%, indicating a well-balanced performance between precision and recall. The ROC AUC of 94.53% demonstrates strong ability to differentiate Normal from Abnormal cases. Precision was higher for Normal (90%) than Abnormal (85%), while recall was slightly better for Abnormal (89%) than Normal (87%), reflecting the model's slightly greater sensitivity to Abnormal cases. The F1-scores of 0.89 and 0.87 indicate consistent predictive effectiveness across both classes. Macro and weighted averages for all metrics were 0.88, indicating reliable overall performance despite some class imbalance. The best hyperparameters included a maximum depth of 10, minimum samples per leaf of 4, and minimum samples to split of 2. These findings are summarised in Figure 5.15, with the ROC curve illustrated in Figure 5.16.

5.6 Result for Logistic Regression

```
Training set evaluation:
Accuracy: 0.7228
F1-score: 0.6999
ROC AUC: 0.7815
              precision    recall  f1-score   support

      Normal       0.76      0.72      0.74      1006
    Abnormal       0.68      0.72      0.70       816

    accuracy                           0.72      1822
   macro avg       0.72      0.72      0.72      1822
weighted avg       0.73      0.72      0.72      1822
```

Figure 5.17: Performance of Logistic Regression on Training Set

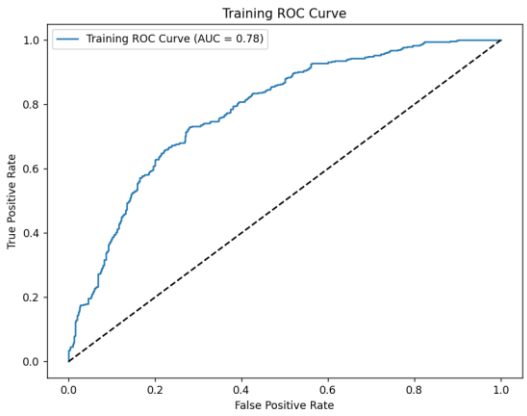


Figure 5.18: ROC Curve of Logistic Regression on Training Set

The Logistic Regression model's evaluation on the training set shows an overall accuracy of 72.28%, with an F1-score of 69.99%, indicating a moderate balance between precision and recall. The ROC AUC of 78.15% reflects a fair capability to discriminate between the Normal and Abnormal classes. Class-specific results reveal higher precision for the Normal class (76%) than the Abnormal class (68%), while recall is equal for both classes at 72%. As a result, the F1-score is slightly higher for Normal (0.74) compared to Abnormal (0.70), suggesting a somewhat better performance in identifying Normal instances. Macro averages for precision, recall, and F1-score are all 0.72, showing equal weight given to both classes, whereas weighted averages, which consider class imbalance, are marginally higher at approximately 0.72 to 0.73. Overall, these metrics indicate moderate but reliable classification performance by the Logistic Regression model on the training data. These results are illustrated in Figure 5.17 and Figure 5.18.

```
Test set evaluation:
Accuracy: 0.7270
F1-score: 0.6952
ROC AUC: 0.7849
              precision    recall  f1-score   support

      Normal       0.75      0.75      0.75       216
    Abnormal       0.70      0.69      0.70       176

    accuracy                           0.73       392
   macro avg       0.72      0.72      0.72       392
weighted avg       0.73      0.73      0.73       392

Best Parameters: {'C': 0.1, 'max_iter': 100, 'solver': 'liblinear'}
```

Figure 5.19: Performance of Logistic Regression on Test Set

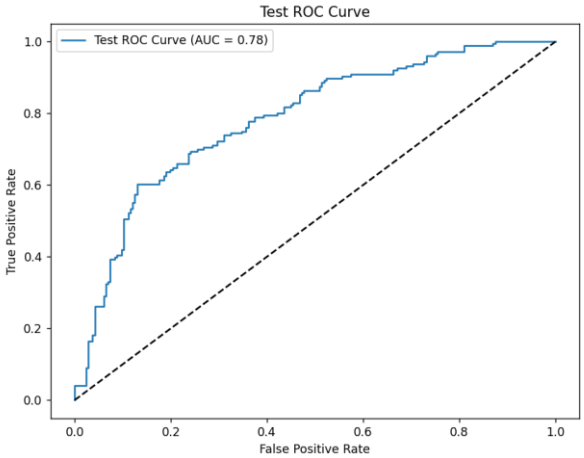


Figure 5.20: ROC Curve of Logistic Regression on Test Set

The Logistic Regression model's performance on the test set achieved an accuracy of 72.70%, with an F1-score of 69.52%, indicating a moderate balance between precision and recall. The ROC AUC of 78.49% suggests the model maintains reasonable ability to differentiate between Normal and Abnormal cases. Precision was higher for Normal (75%) compared to Abnormal (70%), while recall values were 75% for Normal and 69% for Abnormal. This results in F1-scores of 0.75 for Normal and 0.70 for Abnormal, showing slightly better performance on the Normal class. Both macro and weighted averages for precision, recall, and F1-score hover around 0.72–0.73, reflecting balanced performance considering class distribution. The best hyperparameters found were C=0.1, max_iter=100, and solver='liblinear'. These evaluation metrics are summarized in Figure 5.19 and Figure 5.20.

5.7 Comparison of Result

This section presents a comparative analysis of five machine learning models including Support Vector Machine (SVM), Random Forest, XGBoost, Decision Tree, and Logistic Regression. The comparison is based on key evaluation metrics including accuracy, F1-score, and AUC (Area Under the ROC Curve), assessed on both the training and test datasets.

Based on Table 5.1 and Table 5.2, the comparison of results of the models on both the training and test sets demonstrates differences in performance in predicting cervical cancer using the colposcopy image dataset.

Table 5.1: Summary of Training Set Modelling Result

Model	Accuracy (%)	F1-Score (%)	AUC (%)
SVM	71.41	69.90	79.15
Random Forest	98.68	98.54	99.95
XGBoost	76.51	72.14	85.37
Decision Tree	92.26	91.55	98.38
Logistic Regression	72.28	69.99	78.15

Table 5.2: Summary of Test Set Modelling Result

Model	Accuracy (%)	F1-Score (%)	AUC (%)
SVM	71.94	69.78	78.84
Random Forest	96.17	95.84	99.46
XGBoost	75.00	70.66	83.36
Decision Tree	87.76	86.67	94.53
Logistic Regression	72.70	69.52	78.49

Based on the results in Table 5.1, the Random Forest model demonstrated the highest performance during training, achieving an accuracy of 98.68%, an F1-score of 98.54%, and an AUC of 99.95%, which reflects its strong ability to fit the training data effectively. The Decision Tree also produced solid results, with both accuracy and F1-score exceeding 90%, accompanied by an AUC of 98.38%. On the other hand, XGBoost and Logistic Regression showed moderate effectiveness, with accuracy rates between 72% and 76% and F1-scores below 73%. Notably, SVM recorded the lowest training performance, with an accuracy of 71.41%, an F1-score of 69.90%, and an AUC of 79.15%, indicating comparatively weaker learning on the training dataset.

The evaluation on the test set, as shown in Table 5.2, reveals similar trends. Random Forest maintained its high performance, with an accuracy of 96.17%, an F1-score of 95.84%, and an AUC of 99.46%, highlighting its strong ability to generalize to unseen data. Although the Decision Tree experienced a slight reduction in performance on the test set, it still maintained respectable accuracy and F1-score values of 87.76% and 86.67%, respectively. XGBoost and Logistic Regression continued to deliver moderate predictive results, with metrics hovering around the 70% range. SVM again exhibited the lowest test results, recording an accuracy of 71.94%, an F1-score of 69.78%, and an AUC of 78.84%, reflecting limited effectiveness in correctly classifying new samples.

The relatively poor performance of SVM can be attributed to its sensitivity to the data characteristics, particularly when the features are not easily separable by a linear or simple nonlinear decision boundary. Without precise hyperparameter tuning or the use of an appropriate kernel, SVM may face difficulties in capturing complex classification boundaries. Additional factors, such as inadequate feature scaling and class imbalance, may further affect its predictive capability. In contrast, the enhanced accuracy and reliability of Random Forest

result from its ensemble method, which aggregates numerous decision trees to reduce overfitting and strengthen model stability. Its capacity to model nonlinear relationships and handle diverse feature types makes it particularly well-suited for this classification problem.

Overall, Random Forest has demonstrated itself as the most effective and reliable model for classifying cervical cancer in this study. Its consistent accuracy across training and test datasets, with only slight decreases, indicates it has successfully established a generalised decision boundary that performs well on unseen data.

5.8 Summary

This chapter presented a comparative evaluation of five machine learning models for cervical cancer classification based on colour features extracted from colposcopy images. Random Forest consistently achieved the highest accuracy and demonstrated strong generalisation across both training and test datasets. Decision Tree showed reasonable performance but with a noticeable decline on unseen data. XGBoost and Logistic Regression delivered moderate results, while SVM obtained the lowest performance in both training and testing phases. Overall, the results highlight Random Forest as the most reliable and effective model for accurate cervical cancer classification in this study.

Chapter 6

Conclusion and Future Works

6.1 Introduction

In this chapter, the conclusions drawn from this study are presented. The objectives were to evaluate the performance of various machine learning algorithms and determine the most effective model for accurately classifying cervical cancer. This was achieved by applying a combination of colour-based feature extraction techniques and machine learning algorithms to colposcopy image datasets. This chapter summarises the key findings, outlines the contributions of the study, discusses its limitations, and provides suggestions for future work.

6.2 Contributions

This study contributes to the classification of cervical cancer by applying colour-based feature extraction techniques to colposcopy images and evaluating their performance using various machine learning algorithms. The extracted colour features include the mean and standard deviation of RGB values, as well as the dominant colour of each image. These features were chosen for their simplicity and efficiency in capturing visual patterns that may differentiate between normal and abnormal cervical tissues.

Five machine learning algorithms including Support Vector Machine (SVM), Random Forest, XGBoost, Decision Tree, and Logistic Regression were trained and evaluated based on the extracted colour information. The study carefully compared the models' performance based on accuracy, F1-score, and AUC to determine their ability to classify cervical conditions correctly.

The classification results revealed that the SVM model achieved the highest scores across key metrics on the test dataset, demonstrating strong generalisation and effectiveness in

detecting abnormal cervical conditions. This finding highlights the potential of combining colour feature extraction with machine learning algorithms to support early detection of cervical cancer and help medical professionals in diagnosis.

The following table summarises how the study's objectives were achieved:

Table 6.1: Summary of Study Objectives and Their Achievements

Objectives	Achievements
To extract colour features for colposcopy images for the classification of cervical cancer.	Successfully extracted mean, standard deviation of RGB values, and the dominant colour from the images.
To evaluate the performance of the machine learning models for classifying cervical cancer	Trained and tested SVM, Random Forest, XGBoost, Decision Tree, and Logistic Regression models.
To determine the best model for the classification of cervical cancer.	Identified Random Forest as the best-performing model based on accuracy, F1-score, and AUC evaluation metrics.

6.3 Limitations

One of the major limitations of this study is the insufficient availability of colposcopy image datasets. The dataset used was relatively small and lacked diversity in terms of image sources, cervical conditions, and patient backgrounds. Limited data makes it challenging to capture the full range of variations found in real clinical cases, which may affect the consistency and reliability of the model's performance. A broader dataset could provide a better foundation for training models that need to perform well across different clinical scenarios.

Another limitation is the exclusive use of colour features derived only from the RGB colour space. The selected features, including the mean and standard deviation of RGB values as well as the dominant colour, were chosen for their simplicity and computational efficiency. However, relying solely on basic colour information may not fully capture the complex visual patterns present in colposcopy images. Important visual characteristics related to texture, structural differences, or spatial arrangements might be overlooked, which could reduce the model's ability to accurately distinguish between various cervical conditions.

In addition, the study only explored traditional machine learning models such as Support Vector Machine, Random Forest, XGBoost, Decision Tree, and Logistic Regression. Although these algorithms are effective for structured data and relatively simple feature sets, they may not be the best choice for dealing with complex image data. The study did not investigate deep learning approaches, which are known for their ability to automatically learn detailed and high-level features from raw image inputs. This limitation may have restricted the study from achieving higher accuracy or capturing more complex visual information that could improve classification results.

6.4 Future Works

To address the limited dataset availability, future work could focus on gathering a larger and more diverse collection of colposcopy images. Collaborating with hospitals, research institutions, or public health organisations may help expand the dataset and ensure it includes images from various sources, populations, and imaging conditions. A broader dataset would improve the generalisability of the models and allow for more reliable performance evaluations across different clinical scenarios.

Beyond relying solely on RGB-based colour features, future studies could expand feature extraction by incorporating additional visual attributes. This includes integrating texture-based features such as Gray-Level Co-occurrence Matrix (GLCM) or Local Binary Patterns (LBP), and exploring other colour spaces like HSV, LAB, or YCbCr that may offer a more accurate representation of tissue appearance. Combining these features could lead more informative dataset for model training.

Lastly, future work could explore deep learning methods, particularly convolutional neural networks (CNNs), which are well-suited for image analysis tasks. Deep learning models can automatically learn hierarchical features from raw images, potentially capturing subtle and complex patterns that machine learning models might miss. With sufficient training data and computational resources, deep learning approaches could further enhance the accuracy and clinical applicability of cervical cancer detection tools.

6.5 Conclusion

This study contributes to cervical cancer detection by applying colour-based feature extraction from colposcopy images combined with machine learning classifiers, demonstrating that simple features such as the mean, standard deviation, and dominant RGB colours can effectively distinguish between normal and abnormal cervical tissues. Among the models evaluated, the Random Forest achieved the best overall performance in terms of accuracy, F1-score, and AUC, highlighting its potential for reliable classification. However, limitations including a relatively small dataset, exclusive reliance on RGB colour features without incorporating texture or spatial information, and the absence of advanced deep learning approaches indicate areas for improvement. Despite these limitations, the findings highlight the potential of combining colour features with machine learning for cervical cancer classification. Future work involving larger datasets, more diverse feature extraction, and the

integration of deep learning techniques could further improve diagnostic accuracy and clinical relevance, ultimately supporting the development of effective automated screening tools.

References

Arbyn, M., Weiderpass, E., Bruni, L., De Sanjosé, S., Saraiya, M., Ferlay, J., & Bray, F. (2019). Estimates of incidence and mortality of cervical cancer in 2018: a worldwide analysis. *The Lancet Global Health*, *8*(2), e191–e203. https://doi.org/10.1016/s2214-109x(19)30482-6

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/a:1010933404324

Chen, T., & Guestrin, C. (2017). XGBoost: A Scalable Tree Boosting System. *ACM*, 13–17. https://doi.org/10.1145/2939672.2939785

Cooper, D. B., & Dunton, C. J. (2023). *Colposcopy*. StatPearls - NCBI Bookshelf. https://www.ncbi.nlm.nih.gov/books/NBK564514/

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. https://doi.org/10.1007/bf00994018

Demajo, L. M. (2020). Explainable AI for Interpretable Credit Scoring. *University of Malta*. https://www.researchgate.net/publication/350874464_Explainable_AI_for_Interpretable_Credit_Scoring

Devi, S., Komalavalli, C., & Chinnaiyan, R. (2023). *Machine learning based classification models for early analysis and prediction of cervical cancer*. IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/10526129

Dhaliwal, S. S., Nahid, A., & Abbas, R. (2018). Effective intrusion detection system using XGBOOST. *Information*, *9*(7), 149. https://doi.org/10.3390/info9070149

Fu, H., & Qi, K. (2022). Evaluation Model of Teachers' Teaching Ability Based onImproved Random Forest with Grey Relation Projection. *Scientific Programming*, *2022*(4), 1–12. https://10.1155/2022/5793459

Ganguly, T., Pati, P. B., Deepa, K., Singh, T., & Özer, T. (2023). *Machine Learning based Comparative Analysis of Cervical Cancer Risk Classifications Algorithms*. IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/10200617

Garland, S., Park, S., Ngan, H., Frazer, I., Tay, E., Chen, C., Bhatla, N., Pitts, M., Shin, H., Konno, R., Smith, J., Pagliusi, S., & Park, J. (2008). The need for public education on HPV and cervical cancer prevention in Asia. *Vaccine*, *26*(43), 5435–5440. https://doi.org/10.1016/j.vaccine.2008.07.077

Grandini, M., Bagli, E., & Visani, G. (2020). Metrics for multi-class Classification: An overview. *A WHITE PAPER*, 1–3. https://arxiv.org/pdf/2008.05756

Hosmer, D. W., Jr, Lemeshow, S., & Sturdivant, R. X. (2013). Applied Logistic Regression (3rd ed.). Wiley-Interscience.

Hull, R., Mbele, M., Makhafola, T., Hicks, C., Wang, S., Reis, R., Mehrotra, R., Mkhize-Kwitshana, Z., Kibiki, G., Bates, D., & Dlamini, Z. (2020). Cervical cancer in low and middle-income countries (Review). *Oncology Letters*, *20*(3), 2058–2074. https://doi.org/10.3892/ol.2020.11754

HPV Information Centre. (2023). *Malaysia Human Papillomavirus and Related Cancers, Fact Sheet 2023*. https://hpvcentre.net/statistics/reports/MYS_FS.pdf

Jha, M., Gupta, R., & Saxena, R. (2021). *Cervical cancer risk prediction using XGBoost classifier*. IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/9673474

Kashyap, N., Krishnan, N., Kaur, S., & Ghai, S. (2019). Risk factors of Cervical Cancer: A Case-Control Study. *Asia-Pacific Journal of ONCOLOGY NURSING*, *6*(3), 308–314. https://doi.org/10.4103/apjon.apjon_73_18

Kulkrani, A., Batarseh, F. A., & Chong, D. (2021). Foundations of Data Imbalance and Solutions for a Data Democracy. *A White Paper*. https://arxiv.org/pdf/2108.00071

Loopik, D. L., IntHout, J., Melchers, W. J. G., Massuger, L. F. A. G., Bekkers, R. L. M., & Siebers, A. G. (2018). Oral contraceptive and intrauterine device use and the risk of cervical intraepithelial neoplasia grade III or worse: a population-based study. *European Journal of Cancer, 124*, 102–109. https://doi.org/10.1016/j.ejca.2019.10.009

Louppe, G. (2012). Understanding Random Forests: From Theory to Practice. *University of Liège*. https://10.13140/2.1.1570.5928

Mahesh, B. (2018). Machine Learning Algorithms - A Review. *International Journal of Science and Research (IJSR), 9*(1), 381–386. https://10.21275/ART20203995

Marios, C. A. (2023). *What is decision tree learning in machine learning?* Train Test Split | Explorations of Machine Learning, AI, and Data Science. https://traintestsplit.com/what-is-decision-tree-learning-in-machine-learning/

Miller, C. (2018). *Hands-On Data Analysis with NumPy and Pandas: Implement Python Packages from Data Manipulation to Processing*. Packt Publishing.

National Cancer Institute. (2023). *What is cervical cancer?* National Institutes of Health. https://www.cancer.gov/types/cervical#:~:text=Cervical%20cancer%20is%20cancer%20that,cervix%20and%20to%20surrounding%20areas.

National Cancer Society Malaysia. (2018). *Cervical Cancer Screenings.* https://cancer.org.my/get-screened/cancer-screening-clinics/cervical-cancer-screening/

Nersesyan, A., Muradyan, R., Kundi, M., Fenech, M., Bolognesi, C., & Knasmueller, S. (2020). Smoking causes induction of micronuclei and other nuclear anomalies in cervical cells. *International Journal of Hygiene and Environmental Health*, *226*, 113492. https://doi.org/10.1016/j.ijheh.2020.113492

Nobel, S. M. N., Sultana, S., Singha, S. P., Chaki, S., Mahi, M. J. N., Jan, T., Barros, A., & Whaiduzzaman, M. (2024). Unmasking banking fraud: unleashing the power of

machine learning and explainable AI (XAI) on imbalanced data. *Information*, *15*(6), 298. https://doi.org/10.3390/info15060298

Prasher, S., Nelson, L., & Sandhya, V. (2023). *Early diagnosis of cervical cancer using Machine Learning approaches: Comparative Analysis*. IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/10368874

Probst, P., Wright, M. N., & Boulesteix, A. (2019). Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews Data Mining and Knowledge Discovery*, *9*(3). https://doi.org/10.1002/widm.1301

Rahman, Md. A., & Ghosh, A. (2023). *Risk factor analysis and prediction of Cervical Cancer based on machine learning models*. IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/10441011

Rawat, P., Bajaj, M., Mehta, S., Sharma, V., & Vats, S. (2023). *A Study on Cervical Cancer Prediction using Various Machine Learning Approaches*. IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/10099493

Rezende, M. T., Bianchi, A. G. C., & Carneiro, C. M. (2021). Cervical cancer: Automation of Pap test screening. *Diagnostic Cytopathology*, *49*(4), 559–574. https://doi.org/10.1002/dc.24708

Rougier, N. P. (2021). *Scientific Visualization: Python + Matplotlib*. Nicolas P. Rougier. https://inria.hal.science/hal-03427242/document

Santhanam, R., Uzir, N., Raman, S., & Banerjee, S. (2016). Experimenting XGBoost Algorithm for Prediction and Classification of Different Datasets. *International Journal of Control Theory and Applications*, *9*(40), 651–662. https://www.researchgate.net/publication/318132203

Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., & Williamson, R. C. (2001). Estimating Support of a High-Dimensional Distributio. *Neural Computation*, *13*(7), 1443–1471. https://10.1162/089976601750264965

Seng, L. M., Rosman, A. N., Khan, A., Haris, N. M., Mustapha, N. a. S., Husaini, N. S. M., & Zahari, N. F. (2018). *Awareness of cervical cancer among women in Malaysia*. https://pmc.ncbi.nlm.nih.gov/articles/PMC6040851/#ref-list1

Shanmugamani, R. (2018). *Deep Learning for Computer Vision: Expert techniques to train advanced neural networks using Tensorflow and KERAS*. Packt Publishing.

Soğukkuyu, D. Y. C., & Ata, O. (2022). *Diagnosing cervical cancer using machine learning methods*. IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/9800033

Srivastav, S., Guleria, K., & Sharma, S. (2023). *Predictive Machine Learning Approaches for Cervical Cancer Detection: An analytical comparison*. IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/10112553

Sundarambal, B., Karthikeyini, C., Bommi, R. M., Subramanian, S., & Jacintha, V. (2022). *Comparison of Machine Learning and Deep Learning models for Cervical Cancer Prediction*. IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/9780705

Uddin, K. M. M., Mamun, A. A., Chakrabarti, A., Mostafiz, R., & Dey, S. K. (2024). An ensemble machine learning-based approach to predict cervical cancer using hybrid feature selection. *Neuroscience Informatics*, *4*(3). https://doi.org/10.1016/j.neuri.2024.100169

Van Otten, N. (2024). *Support Vector Machines (SVM) In Machine Learning Made Simple & How To Tutorial*. Spot Intelligence. https://spotintelligence.com/2024/05/06/support-vector-machines-svm/

VIA. (2024). *VIA 2500 image Classification Dataset*. Roboflow. https://universe.roboflow.com/via-test/via-2500-image

World Health Organization: WHO. (2024). Cervical cancer. https://www.who.int/news-room/fact-sheets/detail/cervical-cancer

Yilmaz, Ö. (2023). From the Roots to the Final Branches: A Journey into the Depths of Decision Tree Algorithms. *Medium.* https://medium.com/@omrylmzz35/from-the-roots-to-the-final-branches-a-journey-into-the-depths-of-decision-tree-algorithms-208ff3d97d7e

Appendix

Appendix A: Access to Model Demo

The interactive model is hosted on Hugging Face Spaces and can be accessed at:

https://huggingface.co/spaces/ssyafiqahsiti/RandomForest_cervical_cancer